

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Харківський національний університет радіоелектроніки  
Факультет Комп'ютерних наук  
Кафедра Програмної інженерії

## КВАЛІФІКАЦІЙНА РОБОТА

### Пояснювальна записка

\_\_\_\_\_ другий (магістерський) \_\_\_\_\_

(рівень вищої освіти)

\_\_\_\_\_ Дослідження методів прогнозування динаміки \_\_\_\_\_  
\_\_\_\_\_ курсу криптовалют \_\_\_\_\_

Виконав:

студент 2 курсу, групи ПЗМ-21-2

\_\_\_\_\_ Авдєєв О. Д. \_\_\_\_\_

(прізвище, ініціали)

Спеціальність 121 – Інженерія програмного

\_\_\_\_\_ забезпечення \_\_\_\_\_

Тип програми освітньо-наукова

Керівник доц. Каук В. І.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри \_\_\_\_\_

(підпис)

\_\_\_\_\_ З.В. Дудар \_\_\_\_\_

(прізвище, ініціали)

2023 р.

## Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ Комп'ютерних наук \_\_\_\_\_  
Кафедра \_\_\_\_\_ Програмної Інженерії \_\_\_\_\_  
Рівень вищої освіти \_\_\_\_\_ другий (магістерський) \_\_\_\_\_  
Спеціальність \_\_\_\_\_ 121 – Інженерія програмного забезпечення \_\_\_\_\_  
(код і повна назва)  
Тип програми \_\_\_\_\_ освітньо-наукова програма \_\_\_\_\_  
Освітня програма \_\_\_\_\_ Інженерія програмного забезпечення \_\_\_\_\_  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

«\_\_\_» \_\_\_\_\_ 202\_ р.

**ЗАВДАННЯ**

## НА КВАЛІФІКАЦІЙНУ РОБОТУ

студента \_\_\_\_\_ Авдєєву Олексію Дмитровичу \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи \_\_\_\_\_ «Дослідження методів прогнозування динаміки курсу  
криптовалют» \_\_\_\_\_

затверджена наказом університету від « 29 » березня 2023 р. № 302 Ст

2. Термін подання студентом до екзаменаційної комісії « 16 » травня 2023 р.

3. Вихідні дані до роботи \_\_\_\_\_ *встановлений календарний план роботи,  
методичні вказівки до оформлення пояснювальної записки, методи прогнозування  
засновані на авторегресії та нейронних мережах.* \_\_\_\_\_

4. Перелік питань, що потрібно опрацювати в роботі \_\_\_\_\_ *аналіз предметної  
галузі, постановка задачі, обґрунтування методів дослідження, етапи проведення  
наукового дослідження, математичний апарат, методологія проведення  
дослідження, реалізація обраних моделей, проведення експерименту та аналіз  
результатів дослідження.* \_\_\_\_\_

## КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз предметної галузі	29.03.2023	виконано
2	Огляд методів та етапів дослідження	31.03.2023	виконано
3	Огляд математичного апарату	02.04.2023	виконано
4	Огляд експериментальної складової дослідження та побудова плану експерименту	05.04.2023	виконано
5	Імплементация обраних моделей	06.04.2023	виконано
6	Проведення експерименту та аналіз результатів дослідження	10.04.2023	виконано
7	Написання пояснювальної записки	17.04.2023	виконано
8	Перевірка пояснювальної записки керівником, підготовка роботи для проходження перевірки на антиплагіат та проходження нормконтролю	05.05.2023	виконано
9	Оцінка роботи рецензентом, отримання відзиву від керівника кваліфікаційної роботи, попередній захист роботи	09.05.2023	виконано
10	Здача роботи у електронний архів, допуск роботи до захисту завідувачем кафедри та передача готової роботи секретарю ЕК	13.05.2023	виконано
11	Захист атестаційної роботи	16.05.2023	виконано

Дата видачі завдання 29 березня 2023 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_ доц. Каук В.І.  
(підпис) (посада, прізвище, ініціали)

**РЕФЕРАТ / ABSTRACT**

Кваліфікаційна робота магістра містить: 87 с., 16 рис., 11 табл., 32 джер.

КРИПТОВАЛЮТИ, АВТОРЕГРЕСІЯ, ЧАСОВИЙ РЯД, ARMA, ARIMA, LSTM, GRU, ПРОГНОЗУВАННЯ, АНАЛІЗ, ДОСЛІДЖЕННЯ.

Об'єктом дослідження є алгоритми сімейства авторегресії та рекурентних нейронних мереж для прогнозування динаміки курсу криптовалют.

Метою роботи є проведення дослідження щодо ефективності алгоритмів сімейства авторегресії та рекурентних нейронних мереж для прогнозування динаміки курсу криптовалют.

У результаті роботи було досліджено та реалізовано алгоритми сімейства авторегресії ARMA, ARIMA та рекурентних нейронних мереж LSTM, GRU. Був проведений експеримент для порівняння реалізованих алгоритмів за заданими критеріями та визначена найбільш ефективна модель для прогнозування динаміки курсу криптовалют.

CRYPTOCURRENCIES, AUTOREGRESSION, TIME SERIES, ARMA, ARIMA, LSTM, GRU, FORECASTING, ANALYSIS, RESEARCH.

The object of the study is the algorithms of the autoregression family and recurrent neural networks for forecasting the dynamics of the course of cryptocurrencies.

The purpose of the work is to conduct the research on the effectiveness of algorithms of the autoregression family and recurrent neural networks for forecasting the dynamics of the course of cryptocurrencies.

As a result of the work, the algorithms of the autoregression family ARMA, ARIMA and recurrent neural networks LSTM, GRU were studied and implemented. An experiment was conducted to compare implemented algorithms according to the specified criteria and the most effective model for forecasting the dynamics of the course was determined.

## Умови публікації пояснювальної записки

Я, Авдєєв Олексій Дмитрович  
(прізвище, ім'я, по батькові)  
студент групи ПЗМ-21-2 здобувач вищої освіти на другому (магістерському) рівні  
кафедра програмної інженерії,  
(повна назва кафедри)  
заявляю: моя кваліфікаційна робота на тему

Дослідження методів прогнозування динаміки курсу криптовалют  
(назва роботи)

що буде представлена до ЕК для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу EIArKhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений (а) з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

## ЗМІСТ

Вступ.....	8
1 Опис проблемної галузі .....	10
1.1 Аналіз предметної області.....	10
1.2 Постановка задачі.....	14
2 Методи та етапи дослідження.....	15
2.1 Обґрунтування методів дослідження.....	15
2.2 Етапи проведення наукового дослідження .....	15
3 Математичний апарат .....	16
3.1 Моделі ARMA та ARIMA .....	16
3.2 Рекурентні нейронні мережі .....	18
3.3 Нейронна мережа типу LSTM.....	20
3.4 Нейронна мережа типу GRU.....	23
4 Експериментальна складова дослідження.....	25
4.1 Вибір криптовалют для дослідження.....	25
4.2 Підготовка даних.....	29
4.3 Нормалізація даних .....	33
4.4 План-програма експерименту .....	34
5 Реалізація моделей .....	36
5.1 Реалізація LSTM та GRU.....	37
5.2 Реалізація ARMA та ARIMA.....	39
6 Проведення експерименту.....	41
6.1 Модель LSTM .....	41
6.2 Модель GRU .....	44
6.3 Модель ARMA.....	47
6.4 Модель ARIMA .....	49
6.5 Аналіз результатів .....	51
6.6 Оптимізація моделей.....	53
6.7 Практична цінність отриманих результатів .....	56

	7
Висновки .....	57
Перелік джерел посилання .....	59
Перелік джерел посилання за науковими напрямками керівника та науковців кафедри програмної інженерії .....	62
Додаток А Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ .....	63
Додаток Б Презентаційні слайди для захисту кваліфікаційної роботи .....	64
Додаток В Текст наукової публікації за темою кваліфікаційної роботи.....	73
Додаток Г Експертний висновок результатів перевірки кваліфікаційної роботи на відповідність оформлення вимоги ДСТУ 3008: 2015.....	80
Додаток Д Програмний код.....	81

## ВСТУП

Криптовалюта вже багато років є модним словом, але останніми роками її популярність різко зросла. Те, що раніше було альтернативним інвестуванням виключно для спекулянтів, тепер відкрито обговорюється як життєздатний варіант для частини портфеля будь-якого інвестора, навіть у його пенсійних планах [1].

Перша половина 2022 року була дуже поганою для крипторинку. Біткоїн та етеріум впали більш ніж на 50% від своїх історичних максимумів наприкінці 2021 року. Незважаючи на те, що останніми тижнями спостерігалися невеликі сплески, крипторинок в цілому в основному зупинився. Хоча ніхто не знає напевно, деякі експерти кажуть, що ціни на криптовалюту можуть впасти ще більше, перш ніж почнеться стійке відновлення [1]. З іншого боку, багато людей стверджують, що у найближчий час знову буде стрімкий рост криптовалют.

У 2021 році біткоїн сягнув кількох нових найвищих цін за весь час, після чого послідувало значне падіння. Етеріум, друга за величиною криптовалюта, також досягла свого нового історичного максимуму наприкінці минулого року, а потім у червні впала нижче 900 доларів США, найнижчого рівня з початку 2021 року [2]. Урядові чиновники США та адміністрація Байдена все частіше виявляють інтерес в нових правилах для криптовалюти.

У той же час інтерес людей до криптовалюти залишається високим: це гаряча тема не лише серед інвесторів, але й у масовій культурі, завдяки всім, починаючи від давніх інвесторів, таких як Ілон Маск, і закінчуючи тим хлопцем із вашої середньої школи на Facebook. Щоб отримати інформацію про поточний стан інвестування в криптовалюту, GOBankingRates провів опитування 1037 дорослих на початку квітня 2022 року. Одним із найбільш вражаючих результатів опитування було те, що понад 83% інвесторів у криптовалюту вказали, що вони інвестували в криптовалюту лише протягом двох років або менше [3].

У зв'язку з цим дослідження методів прогнозування криптовалют та розробки спеціального інструментарію для передбачення на даний момент часу видається вкрай актуальною задачею. Слід зазначити, що у науковому публікаційному

просторі дуже рідко можна зустріти роботи, присвячені цій тематиці. Переважно у них розглядаються питання спрямовані на експертну оцінку поточних і майбутніх перспектив розвитку ринку, що розглядається, або на використання спеціальних методів біржового технічного аналізу, що розкриває особливості та тренди курсових коливань «цифрових грошей».

Прогнозування зміни курсу – дуже нетривіальне завдання. Використання фундаментального аналізу для прогнозування волатильності біржових котирувань неефективне, оскільки курс залежить від економіки певної держави; використання технічного аналізу також неефективне, оскільки неможливо визначити стан ринку через формування попиту та пропозиції за рахунок осіб, готових купити чи продати певні товари або послуги по всьому світу. Одним із дієвих механізмів прогнозування тимчасових фінансових рядів є використання моделей авторегресії ковзного середнього [4] та алгоритмів рекурентних нейронних мереж. Особливу актуальність даний підхід набуває в умовах відсутності на поточний час ефективних інструментів прогнозування курсових коливань подібного роду фінансових активів, що характеризуються нерівномірністю коливань курсів у часі, відсутністю прив'язки до базових активів та високим рівнем спекулятивного попиту.

## 1 ОПИС ПРОБЛЕМНОЇ ГАЛУЗІ

### 1.1 Аналіз предметної області

Криптовалюти стають все більш популярними з багатьох причин. Однією з головних причин є те, що вони не регулюються урядами, тому немає нікого, хто б сказав вам, що ви можете або не можете купувати їх [4]. Це призвело до багатьох спекуляцій щодо майбутньої ціни цих монет і різних прогнозів щодо того, яка монета зросте в ціні більше, ніж інші.

Існує дуже багато веб-сайтів для прогнозування курсу криптовалют, але найбільш популярними є:

- CryptoPredictions.com;
- Coincodex.com;
- TradingBeasts;
- CoinMarketcap;

Розглянемо кожен з зазначених вище систем, аби дізнатися які алгоритми прогнозування вони використовують. Варто зазначити, що алгоритм, на якому базується прогнозування на певному веб-сайті може бути комерційною таємницею компанії.

Почнемо з «CryptoPredictions.com». Це веб-сайт, що був заснований у 2021 році та надає прогнози для тисяч криптовалют. На даний момент існує 12 000 криптоактивів, для яких ви можете контролювати щоденні, місячні та річні прогнози (до 4 років вперед).

Домашня сторінка «CryptoPredictions» (див. рис. 1.1) містить важливі відомості, такі як ринкова капіталізація, 24-годинна відсоткова зміна ціни або щоденний обсяг. Ви також можете побачити тут їх семиденний прогноз і діаграму прогнозів на один рік.

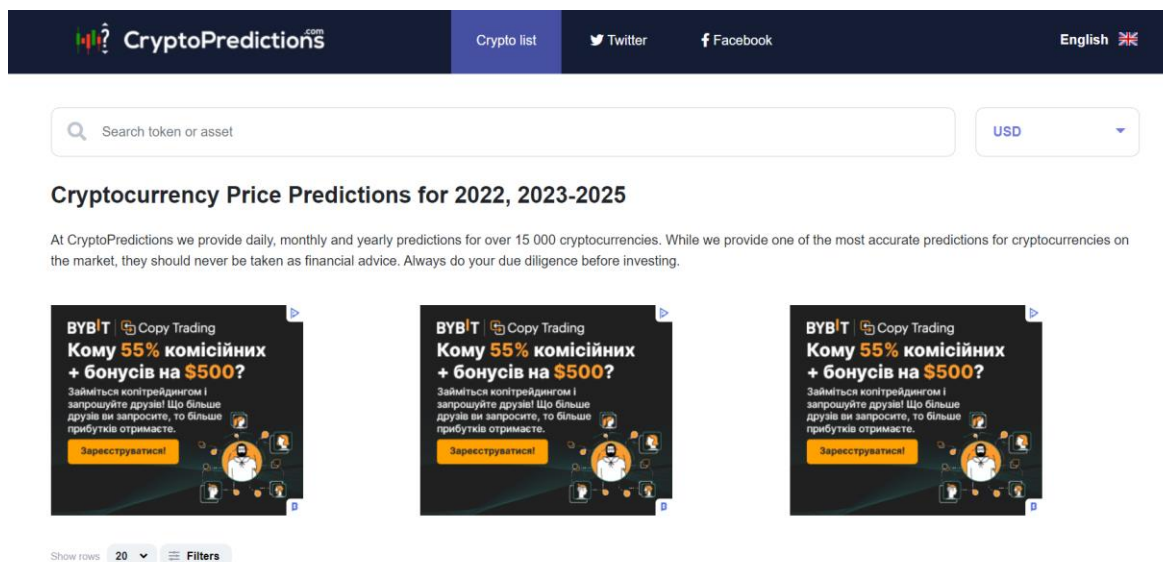


Рисунок 1.1 – Головна сторінка «CryptoPredictions» [5]

Якщо ви хочете побачити більш детальні прогнози, ви можете натиснути назву своєї криптовалюти, і ви перейдете прямо до детальних прогнозів для монети (див. рис. 1.2). На веб-сайті зазначено, що для прогнозування майбутніх цін на криптовалюту вони сканують історичні ринкові дані з бірж. Крім того, вони інформують читачів про те, що віртуальні валюти з довшою історією торгів мають більш якісні прогнози, оскільки вони мають більше даних, які можна використовувати для розрахунку прогнозів.



Рисунок 1.2 – Сторінка прогнозу «CryptoPredictions» [5]

Перейдемо до розгляду веб-сайту «Coincodex.com» (див. рис. 1.3). Він є одним із найпопулярніших веб-сайтів у індустрії криптовалют. Він унікальний,

оскільки надає інформацію майже про кожну монету на ринку, включаючи її ціну, графіки, ринкову капіталізацію та обсяг.

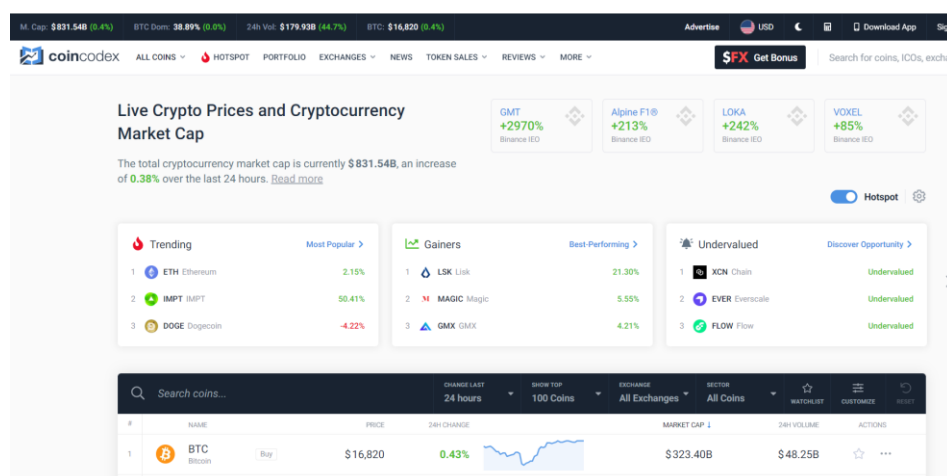


Рисунок 1.3 – Головна сторінка «Coincodex» [6]

Даний ресурс також надає можливість переглядати прогнози понад на 10000 криптовалют на п'ять днів наперед та на один рік, але інформації на яких алгоритмах базуються прогнози у відкритих джерелах не має.

Розглянемо наступний веб-сайт (див. рис. 1.4).

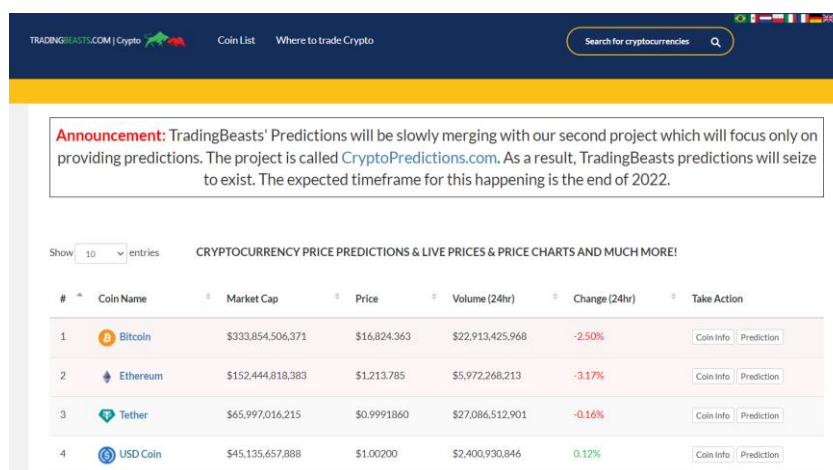


Рисунок 1.4 – Головна сторінка «TradingBeasts» [7]

«TradingBeasts» – це всеохоплюючий веб-сайт, який надає прогнози для понад 2500 монет. За допомогою алгоритму, який працює з історичними даними, використовуючи математичні формули, такі як лінійна регресія або поліноміальні

функції, вони обчислюють, що ймовірно станеться з усіма перерахованими криптоактивами. Веб-сайт надає прогнози на 3 роки вперед, причому кожен місяць містить мінімальні, максимальні та середні прогнозовані ціни. Але через те, що прогноз базується на застарілих алгоритмах, то популярність веб-сайту знижується.

Перейдемо до останньої системи «CoinMarketcap» (див. рис. 1.5).

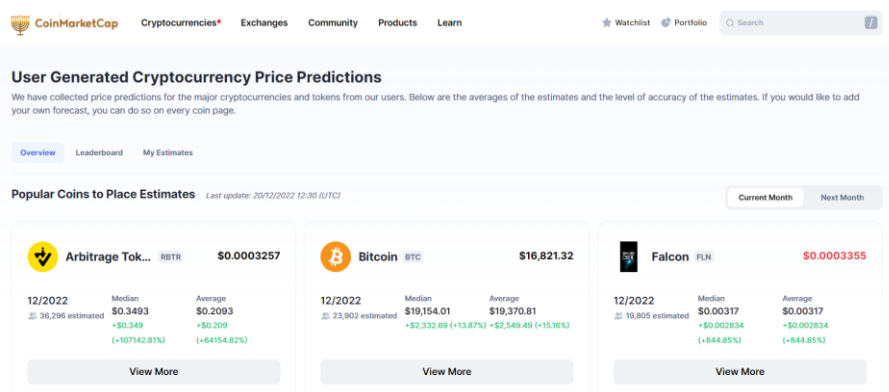


Рисунок 1.5 – Головна сторінка «CoinMarketcap» [8]

«CoinMarketCap» був заснований у 2013 році братами Бренденом і Браяном Чью. Відтоді веб-сайт пройшов довгий шлях і тепер є основним джерелом для всього, що стосується криптовалюти. Крім того, що CoinMarketCap надає актуальні оновлення щодо всіх криптовалют, також містить інформацію про майбутні ICO, новини та події, пов'язані з криптовалютами. У них є оцінки криптовалюти, створені користувачами, монети тут ранжуються на основі кількості оцінок. Система дозволяє вам і всім іншим користувачам надсилати свої оцінки на термін до шести місяців. «Coinmarketcap» збирає всі ці дані в прогноз середньої ціни та медіанного числа. Тобто прогноз даного ресурсу базується лише на думці користувачів, що є досить ненадійним джерелом.

Можемо зробити висновок, що існують популярні веб-ресурси з певним прогнозом для криптовалют, але невідомо на яких алгоритмах він побудований, а саме тому досить складно визначити його точність.

Було проведено додаткове дослідження наукових джерел [9, 10, 11, 12] та за його результатами підтверджено, що у прогнозуванні криптовалют на довгостроковий період часу майже немає сенсу, оскільки ринок дуже волатильний

та залежить від багатьох сторонніх чинників. Також було виявлено, що найбільш вживаними алгоритмами є алгоритми сімейства авторегресії та рекурентних нейронних мереж, а саме:

- LSTM (Long Short-Term Memory): рекурентна нейронна мережа довгої короткострокової пам'яті;
- GRU (Gated Recurrent Unit): рекурентна нейронна мережа;
- ARMA (Autoregressive Moving Average): модель авторегресії рухомого середнього з екзогенними показниками;
- ARIMA (Autoregressive Integrated Moving Average): модель авторегресії інтегрованого рухомого середнього з екзогенними показниками.

## 1.2 Постановка задачі

Проаналізувавши відомі веб-сайти прогнозування курсу криптовалют, можна сказати, що усі системи мають недоліки і є багато методів покращення існуючого функціоналу та додавання нових функцій.

Також було виявлено, що точність прогнозування на довгостроковий період часу є дуже низькою тому було вирішено зосередитися на короткочасному прогнозуванні, а саме на прогнозуванні ціни закриття криптовалюти на поточний день. Прогнозування ціни закриття криптовалюти на поточний день допоможе трейдерам, фокусом яких є ф'ючерсна внутрішньоденна торгівля, краще розуміти коли входити в ордер чи виходити з нього для отримання більшого прибутку.

Метою даної роботи є проведення дослідження для порівняння ефективності застосування алгоритмів сімейства авторегресії ARMA, ARIMA та алгоритмів рекурентних нейронних мереж LSTM, GRU для прогнозування ціни закриття криптовалюти на поточний день.

Для проведення дослідження розглянуті алгоритми будуть реалізовані для прогнозування та будуть визначені переваги та недоліки кожного з них.

## 2 МЕТОДИ ТА ЕТАПИ ДОСЛІДЖЕННЯ

### 2.1 Обґрунтування методів дослідження

Суть дослідження полягає у порівнянні алгоритмів прогнозування, тобто це емпіричне дослідження. Основою емпіричного дослідження є експеримент. Для проведення експерименту необхідно створити стабільне середовище та підготувати тести.

Дослідження виконується з метою знаходження відповіді на поставлене питання, а саме визначенні найкращих методів прогнозування ціни закриття криптовалют для поточного дня.

### 2.2 Етапи проведення наукового дослідження

Головним завданням дослідження є порівняння авторегресійних алгоритмів та рекурентних нейронних мереж задля визначення оптимальних й найкращих з них для моніторингу та прогнозування ринкових змін.

Проведення наукових досліджень буде складатися з наступних етапів:

- дослідження математичних алгоритмів авторегресії та рекурентних нейронних мереж;
- визначення метрик для порівняння;
- реалізація обраних алгоритмів;
- визначення плану експерименту;
- проведення дослідження;
- визначення переваг та недоліків кожного алгоритму, згідно з обраними метриками;
- дослідження шляхів оптимізації моделі;
- аналіз результатів дослідження.

### 3 МАТЕМАТИЧНИЙ АПАРАТ

#### 3.1 Моделі ARMA та ARIMA

В авторегресійній моделі порядку  $p$  (AR( $p$ )) будь-яке поточне значення процесу  $y_t$  виражається як кінцева лінійна сукупність  $p$  попередніх значень процесу та імпульсу  $a_t$  (рівні ряду регресують на своїх попередніх значеннях) [13]:

$$y_t = c + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \dots + \varphi_p y_{t-p} + \varepsilon_t, \quad (3.1)$$

де  $\varphi_1 \dots \varphi_p$  – параметри моделі;

$c$  – константа;

$\varepsilon_t$  – білий шум.

Для простоти константу часто опускають.

Стаціонарний ряд – це ряд, поведінка та властивості якого в теперішньому та майбутньому збігаються з поведінкою в минулому.

Для того, щоб модель була стаціонарною, потрібно накласти деякі обмеження на параметри моделі. Наприклад, при  $|\varphi_1| \geq 1$  модель AR(1) не матиме властивість стаціонарності.

Замість того, щоб використовувати минулі значення змінної прогнозу в регресії, модель ковзного середнього використовує попередні помилки прогнозу в моделі, подібній до регресії [13]. Модель ковзного середнього порядку  $q$  позначається MA( $q$ ) і записується так:

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q}, \quad (3.2)$$

де  $\theta_1 \dots \theta_q$  – параметри моделі;

$c$  – константа;

$\varepsilon_t \dots \varepsilon_{t-q}$  – помилки.

Модель авторегресії рухомого середнього (ARMA) є комбінацією двох моделей: авторегресії порядку  $p$  і рухомого середнього порядку  $q$ . В узагальненому вигляді модель ARMA ( $p, q$ ) виглядає так:

$$y_t = c + \varepsilon_t + \sum_{r=1}^p \varphi_r y_{t-r} + \sum_{r=1}^q \theta_r \varepsilon_{t-r} \quad (3.3)$$

Для нестационарних даних Боксом та Дженкінсом запропоновано модель ARIMA ( $p, d, q$ ), яка після взяття  $d$  послідовних різниць може бути приведена до стаціонарного вигляду, де  $p, d, q$  – структурні параметри, що характеризують порядок для відповідних частин моделі – авторегресійної, інтегрованої та рухомого середнього. Значення  $p$  представляє кількість членів авторегресії,  $d$  представляє кількість разів, коли часовий ряд розрізнявся, а  $q$  представляє кількість членів ковзного середнього. Ці параметри вибираються на основі характеристик часових рядів, що моделюються та результатів статистичних перевірок.

Модель ARIMA дуже схожа на модель ARMA, за винятком того факту, що вона включає ще один фактор, відомий як інтегрований (I), тобто різницю, яка означає I у моделі ARIMA. Коротше кажучи, модель ARIMA – це комбінація ряду відмінностей, уже застосованих до моделі, щоб зробити її стаціонарною та кількості попередніх затримок разом із залишковими помилками для прогнозування майбутніх значень.

Скористаємось інформаційними критеріями для визначення якості моделі. Akaike information criterion (AIC), який можна записати [14]:

$$AIC = 2k - 2\ln(\hat{L}) \quad (3.4)$$

де  $k$  – кількість параметрів у моделі;

$\hat{L}$  – максимальна точність моделі.

Bayesian information criterion (BIC) є ще одним інформаційним критерієм, який можна подати наступним чином:

$$BIC = k \ln(n) - 2 \ln(\hat{L}) \quad (3.5)$$

де  $k$  – кількість параметрів у моделі;

$n$  – розмір набору даних;

$\hat{L}$  – максимальна точність моделі.

Якщо показник інформаційного критерію відносно малий, то модель можна вважати якісною.

### 3.2 Рекурентні нейронні мережі

Рекурентні нейронні мережі (Recurrent Neural Network, RNN) – клас моделей машинного навчання, заснований на використанні попередніх станів мережі для обчислення поточного. Такі мережі зручно застосовувати в тих випадках, коли вхідні дані завдання є нефіксованою послідовністю значень, як, наприклад, текстові дані, де текстовий фрагмент представлений нефіксованою кількістю речень, фраз і слів. Кожен символ у тексті, окремі слова, розділові знаки і навіть цілі фрази – все це може бути атомарним елементом вхідної послідовності [15].

На кожному кроці навчання  $t$  значення  $h^{(t)} \in R^m$  прихованого шару нейронної мережі рекурентної обчислюється наступним чином:

$$h^{(t)} = f(Wx^{(t)} + Uh^{(t-1)} + b^h), \quad (3.6)$$

де  $x^{(t)} \in R^n$  – вхідний вектор у момент часу  $t$  (наприклад, векторне подання поточного слова у текстовому фрагменті);

$W \in R^{mn}, U \in R^{mm}, b^h \in R^m$  – параметри рекурентної нейронної мережі для навчання;

$f$  – функція нелінійного перетворення.

Найчастіше, як нелінійне перетворення застосовують одну з наступних функцій: сигмоїдальна функція (3.7), гіперболічний тангенс (3.8), функція-випрямляч (3.9):

$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}, \quad (3.7)$$

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad (3.8)$$

$$f(x) = \max(0, x) \quad (3.9)$$

Просту рекурентну нейронну мережу можна зобразити наступним чином (див. рис. 3.1):

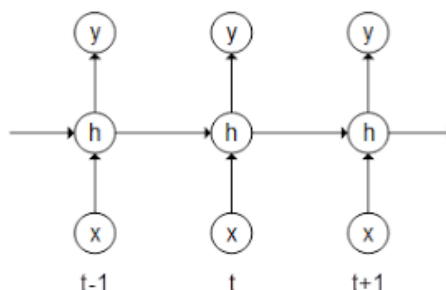


Рисунок 3.1 – Проста рекурентна нейронна мережа (Рисунок виконаний самостійно)

У простій рекурентній нейронній мережі вихідне значення  $y^{(t)}$  на поточному кроці  $t$  обчислюється за формулою:

$$y^{(t)} = W^c h^{(t)} + b^c, \quad (3.10)$$

де  $W^c$  та  $b^c$  – параметри для навчання.

### 3.3 Нейронна мережа типу LSTM

У 1997 році Зепп Хохрайтер та Юрген Шмідхубер представили новий підхід, який отримав назву LSTM (Long Short-Term Memory – довга короткострокова пам'ять). Рекурентні нейронні мережі, засновані на цьому підході, має більш просунутий і складніший спосіб обчислення  $h^{(t)}$  [15]. Даний спосіб, крім вхідних значень та попереднього стану мережі, використовує також фільтри (gates), що визначають, яким чином інформація буде використовуватися для обчислення вихідних значень на поточному шарі  $y^{(t)}$ , так і значень прихованого шару на наступному кроці  $h^{(t+1)}$ . Весь процес обчислення  $h^{(t)}$  для простоти згадується як LSTM шар (LSTM layer, LSTM unit). Архітектура LSTM має вигляд (див. рис. 3.2):

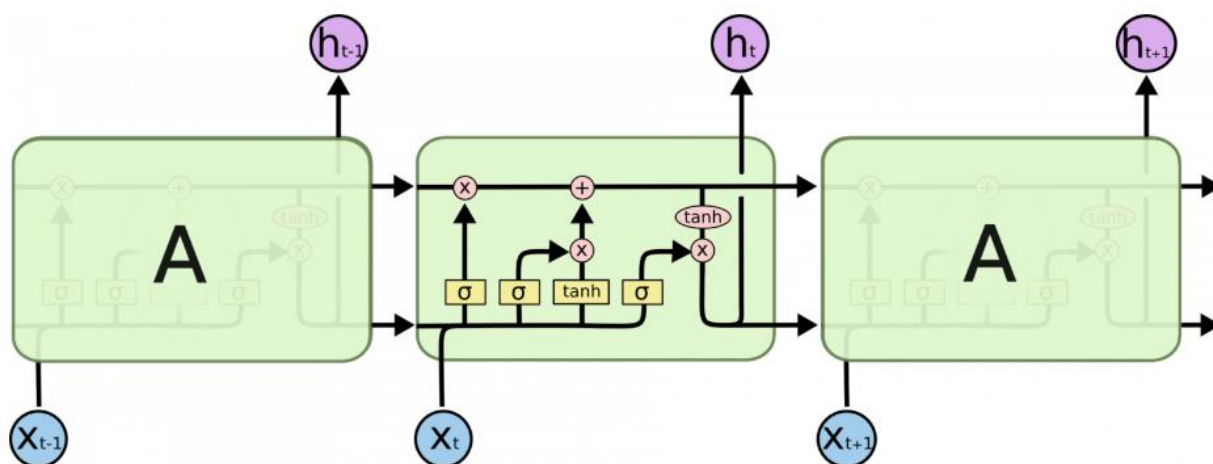


Рисунок 3.2 – Архітектура нейронної мережі LSTM [16]

Розглянемо докладніше структуру LSTM-шару. Центральним поняттям тут є запам'ятовуючий блок (memory cell), який, поряд зі станом мережі  $h$ , обчислюється на кожному кроці, використовуючи поточне вхідне значення  $x^{(t)}$  та значення блоку на попередньому кроці  $c^{(t-1)}$  (див. рис. 3.3).

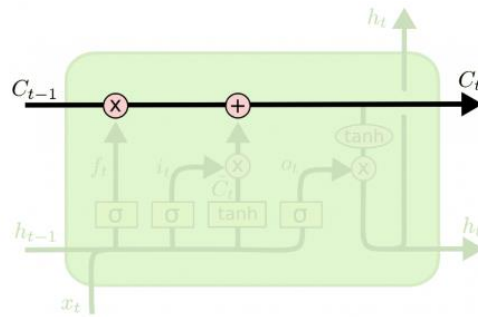


Рисунок 3.3 – Обчислення запам'ятовуючого блоку [16]

LSTM може видаляти інформацію зі стану блоку. Цей процес регулюється структурами, які називаються фільтрами (gates). Фільтри дозволяють пропускати інформацію на підставі деяких умов. Вони складаються з шару сигмоїдальної нейронної мережі та операції поточкового множення. Сигмоїдальний шар повертає числа від нуля до одиниці, які позначають, яку частку кожного блоку інформації слід пропустити далі через мережу [15]. Нуль у даному випадку означає, що не треба пропускати нічого, а одиниця – пропустити все.

Спочатку LSTM визначає, які дані можна виключити зі стану блоку. Це рішення приймає сигмоїдальний шар, званий шаром фільтра забування (forget gate layer). Він дивиться на  $h^{(t-1)}$  і  $x^{(t)}$  і повертає число від 0 до 1 кожного числа зі стану блоку  $c^{(t-1)}$  (див. рис. 3.4).

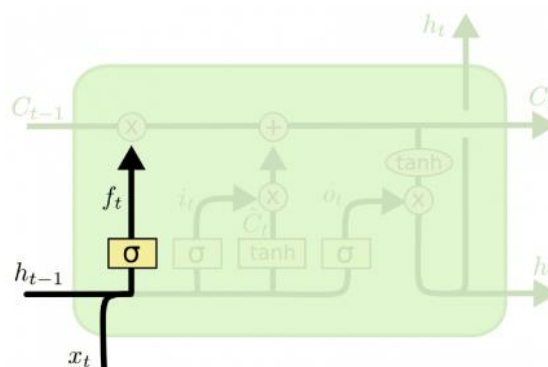


Рисунок 3.4 – Фільтр забування [16]

Сигмоїдальний шар, званий шаром фільтра забування можна описати функцією, що має наступний вигляд:

$$f^{(t)} = \sigma(W^f x^{(t)} + U^f h^{(t-1)}) \quad (3.11)$$

Після цього LSTM вирішує, яка нова інформація буде зберігатися у стані блоку. Цей етап складається з двох частин – вхідного фільтра (input gate) та обчислення стану блоку пам'яті на поточному кроці (див. рис. 3.5) [16].

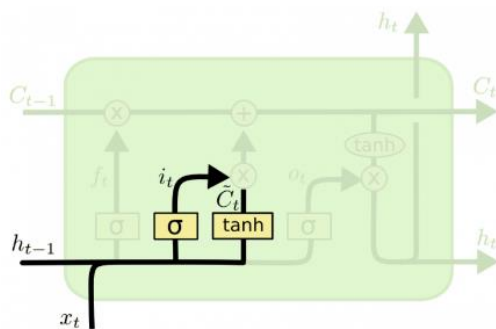


Рисунок 3.5 – Вхідний фільтр та обчислення стану блоку пам'яті на поточному кроці [16]

Вхідний фільтр (input gate)  $i^{(t)}$  визначає, наскільки значення блоку пам'яті на поточному етапі має впливати на результат. Значення фільтра варіюються від 0 (повністю ігнорувати вхідні значення) до 1, що забезпечується областю значень сигмоїдальної функції:

$$i^{(t)} = \sigma(W^i x^{(t)} + U^i h^{(t-1)}) \quad (3.12)$$

Стан блоку пам'яті  $c^{(t)}$  на поточному кроці можна описати наступною функцією:

$$\tilde{c}^{(t)} = \tanh(W^c x^{(t)} + U^c h^{(t-1)}), \quad (3.13)$$

$$c^{(t)} = f^{(t)} \cdot c^{(t-1)} + i^{(t)} \cdot \tilde{c}^{(t)} \quad (3.14)$$

Він (3.14) обчислюється на основі всіх даних, що надходять у момент часу  $t$  (а саме,  $x^{(t)}$ ,  $h^{(t-1)}$  і  $c^{(t-1)}$ ), використовуючи фільтри (3.11) та (3.12) [17].

Далі LSTM вирішує, яку інформацію отримаємо на виході. Вихідні дані будуть засновані на нашому стані блоку та до них буде застосований вихідний фільтр (output gate) [18].

$$o^{(t)} = \sigma(W^o x^{(t)} + U^o h^{(t-1)}) \quad (3.15)$$

Обчислення підсумкового значення зображено нижче (див. рис. 3.6):

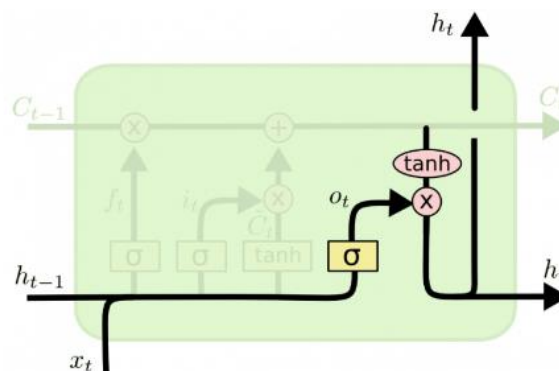


Рисунок 3.6 – Обчислення підсумкового значення [16]

Підсумкове значення LSTM-шару можна описати функцією, що має наступний вигляд:

$$h^{(t)} = o^{(t)} \cdot \tanh(c^{(t)}) \quad (3.16)$$

Підсумкове значення (3.16) визначається вихідним фільтром (3.15) та нелінійною трансформацією над станом блоку пам'яті (3.14):

### 3.4 Нейронна мережа типу GRU

У 2014 році в роботі була представлена модель GRU (Gated Recurrent Unit), заснована на тих же принципах, що і LSTM, але використовує менше фільтрів і

операцій для вичислення  $h^{(t)}$ . Фільтр оновлення (update gate)  $z^{(t)}$  і фільтр скидання стану (reset gate)  $r^{(t)}$  вираховуються за такою формулою:

$$z^{(t)} = \sigma(W^z x^{(t)} + U^z h^{(t-1)}), \quad (3.17)$$

$$r^{(t)} = \sigma(W^r x^{(t)} + U^r h^{(t-1)}) \quad (3.18)$$

Архітектура шару GRU має наступний вигляд (див. рис. 3.7):

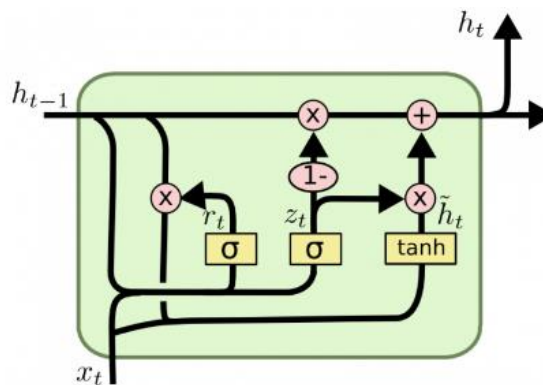


Рисунок 3.7 – Архітектура GRU шару [16]

Вихідне значення вираховується на основі проміжного значення  $\tilde{h}^{(t)}$ , яке, за допомогою фільтра скидання стану (3.18), визначає, які значення попереднього кроку  $h^{(t-1)}$  слід виключити (пряма аналогія з фільтром забуття з LSTM) [19]:

$$\tilde{h}^{(t)} = \tanh(Wx^{(t)} + r^{(t)} + Uh^{(t-1)}) \quad (3.19)$$

Підсумкове значення можна описати функцією, що має наступний вигляд:

$$h^{(t)} = z^{(t)} \cdot h^{(t-1)} + (1 - z^{(t)}) \cdot \tilde{h}^{(t)} \quad (3.20)$$

Підсумкове значення використовує фільтр оновлення (3.17) і проміжне значення (3.19).

## 4 ЕКСПЕРИМЕНТАЛЬНА СКЛАДОВА ДОСЛІДЖЕННЯ

### 4.1 Вибір криптовалют для дослідження

Зараз існує велика кількість криптовалют і наша задача полягає у том, щоб вибрати три надійні криптовалюти з великою капіталізацією, обсягом торгів, соціальною активністю для дослідження математичних алгоритмів авторегресії та рекурентних нейронних мереж [20].

Оберемо наступну множину альтернатив криптовалют:

- 1) XRP;
- 2) Dogecoin;
- 3) Cardano;
- 4) Solana;
- 5) Aave.

Для знаходження найкращих криптовалют з можливих альтернатив визначимо наступні критерії вибору:

- 1) капіталізація – це загальна вартість монет, що знаходяться в обігу [21];
- 2) соціальна активність – обсяги оглядів та обговорень, громадськості, наявність та активність акаунтів у соціальних мережах тощо; також можуть враховуватися кількість результатів запиту в Google;
- 3) час існування криптовалюти (надійність) – криптовалюта має існувати щонайменше рік; це необхідно для отримання принаймні мінімальної впевненості у продовженні її існування у майбутньому;
- 4) обсяг торгів за 24 години – є показником, що використовують для того, щоб зрозуміти як часто торгується конкретний актив та наскільки популярно купувати чи продавати цей конкретний актив у будь-який час; він також може допомогти встановити оптимальний розмір позиції на даному активі, тобто якщо обсяг торгів великий, то це може свідчити про те, що можна відкрити більшу позицію без значного впливу на ціну активу;
- 5) особливості технології – інноваційність, перспективи реального використання.

Розглянемо кожний критерій окремо та визначимо підходящу шкалу оцінок для кожного з них.

Капіталізація та обсяг торгів за добу – це кількісні показники, значення яких можна отримати на веб-ресурсі: <https://coinmarketcap.com/uk/>, який було розглянуто в аналізі предметної області.

Соціальна активність:

– низька – 1 бал: практично повна невідомість монети, відсутність поширеності у соціальних мережах, а також прогнозів та оглядів аналітиків, низька активність на форумах;

– середня – 2 бали: короткострокова популярність, наявність явних перспектив;

– висока – 3 бали: активне обговорення монети на форумах, поширеність у соціальних мережах, велика кількість оглядів та прогнозів, популярність серед блогерів, інтерес з боку ЗМІ.

Час існування криптовалюти:

– низький – 1 бал: менше ніж 5 років;

– середній – 2 бали: від 5 до 8.5 років;

– високий – 3 бали: більше ніж 8.5 років.

Особливості технологій:

– низька – 1 бал: відсутність інноваційності, клони існуючих монет, неясні перспективи, повільні транзакції та високі комісії;

– середня – 2 бали: суміжний стан, наявність як позитивних, так і негативних факторів;

– висока – 3 бали: інноваційна ідея, високі перспективи використання технології у майбутньому. Висока швидкість транзакцій, невеликі комісії, масштабованість.

Після визначення критеріїв та шкали оцінок, перейдемо до моделювання задачі.

Результат моделювання наведено у таблиці 4.1.

Таблиця 4.1 – Моделювання задачі багатокритеріального вибору (Таблиця виконана самостійно)

	Капіталізація (млрд \$)	Соціальна активність	Час існування (у роках)	Обсяг торгів за добу (млн \$)	Особливості технології
XRP	19.6	Компанія активно просуває свою технологію. Велика популярність і ком'юніті	10	801	Активно використовується банками. Не є децентралізованою. Швидкі та дешеві платежі. Висока волатильність. Не є повноцінною криптовалютою
Dogecoin	13.1	Дуже велика поширеність у ЗМІ та на форумах. Величезне інтернет-ком'юніті. Одна із найпопулярніших криптовалют.	9	835	Дуже висока швидкість транзакцій та вкрай низькі комісії. Відсутність принципової інноваційності, однак, перспективи інтеграції з Ethereum.
Cardano	10.9	Дуже швидко увійшла до топ-20 після свого створення. Обговорень та прогнозів не так багато через обмеженість способів купівлі.	8	212	Гарні перспективи. Консенсус між приватністю та вимогами влади. Великі плани у розробників
Solana	4.9	Невелике інтернет-ком'юніті. Постійне розширення числа партнерів та користувачів.	5	250.5	Криптовалюта демонструє пропускну здатність до 60 000 операцій на секунду, безпеку, унікальний алгоритм Proof of History. Також доступні можливості масштабування. Швидкісні показники продовжують зростати.
Aave	0.913	Відсутність поширеності у соціальних мережах, прогнозів та оглядів аналітиків, низька активність на форумах.	2	63.8	Aave базується на Ethereum, комісійні транзакції всередині сервісу залежать від стану мережі ETH, наприклад, її завантаженості.

Наступним кроком замінимо дані на оцінки за шкалою, що було визначено. Результати наведено у таблиці 4.2.

Таблиця 4.2 – Оцінки за визначеною шкалою (Таблиця виконана самостійно)

	Капіталізація (млрд \$)	Соціальна активність	Час існування (у роках)	Обсяг торгів за добу (млн \$)	Особливості технології
XRP	19.6	3	3	801	2
Dogecoin	13.1	2	3	835	2
Cardano	10.9	2	2	212	3
Solana	4.9	2	2	250.5	3
Aave	0.913	1	1	63.8	1

Застосуємо принцип Парето, тобто видалимо одразу заздалегідь найгірший результат, яким у даному випадку є криптовалюта Aave.

Після цього проведемо лінійну адитивну згортку з нормуючими множниками для даної моделі задачі та отримаємо наступні результати (див. табл. 4.3)

Таблиця 4.3 – Результат адитивної згортки (Таблиця виконана самостійно)

	Капіталізація (млрд \$)	Соціальна активність	Час існу- вання (роки)	Обсяг торгів за добу (млн \$)	Особли- вості технології	Результат згортки
Нормую- чий множник	0.02	0.11	0.1	0.0005	0.1	
XRP	19.6	3	3	801	2	1.62
Dogecoin	13.1	2	3	835	2	1.39
Cardano	10.9	2	2	212	3	1.05
Solana	4.9	2	2	250.5	3	0.94

Можемо зробити висновок, що за результатами адитивної згортки трьома найкращими криптовалютами є XRP, Dogecoin та Cardano.

## 4.2 Підготовка даних

Для проведення дослідження у якості джерела даних використаємо один з найвідоміших провайдерів фінансової інформації Yahoo Finance – <https://finance.yahoo.com/> (див. рис. 4.1).

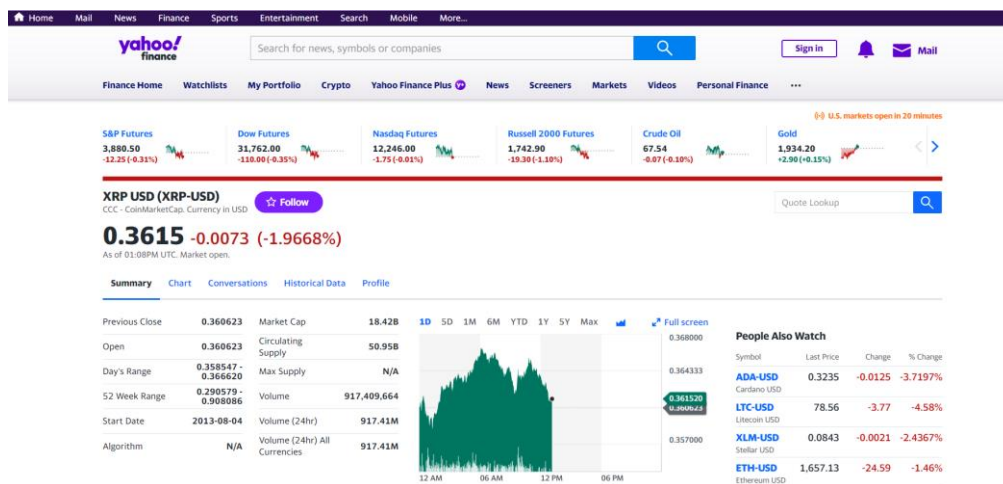


Рисунок 4.1 – Yahoo Finance [22]

Даний ресурс надає безкоштовний API, для якого реалізовані бібліотеки у всіх популярних мовах програмування.

Наступним кроком необхідно визначити який таймфрейм краще всього використовувати для прогнозування. Для цього скористуємось графічним методом кореляційного аналізу часових рядів – побудова діаграми лагів.

Лаг часового ряду – це те, наскільки одна точка відстає в часі від іншої точки даних. У даних часового ряду точки даних позначаються протягом часу з різними інтервалами. Щоб проаналізувати та з'ясувати чи дані часового ряду відповідають будь-якому шаблону, можна використати діаграму лагів.

Діаграма лагів малюється шляхом представлення даних часового ряду на осі абсцис і відставання точки даних часового ряду на осі у. Для точки даних, якщо порядок затримки один, затримка є попередньою точкою даних. Якщо затримка дорівнює двом, затримка є точкою даних перед двома точками даних у часі. За

допомогою діаграми лагів можна шукати закономірності, такі як випадковість, тенденції та сезонність.

Для того щоб побудувати діаграму лагів, оберемо криптовалюту XRP та використаємо мову програмування Python (див. рис. 4.2).

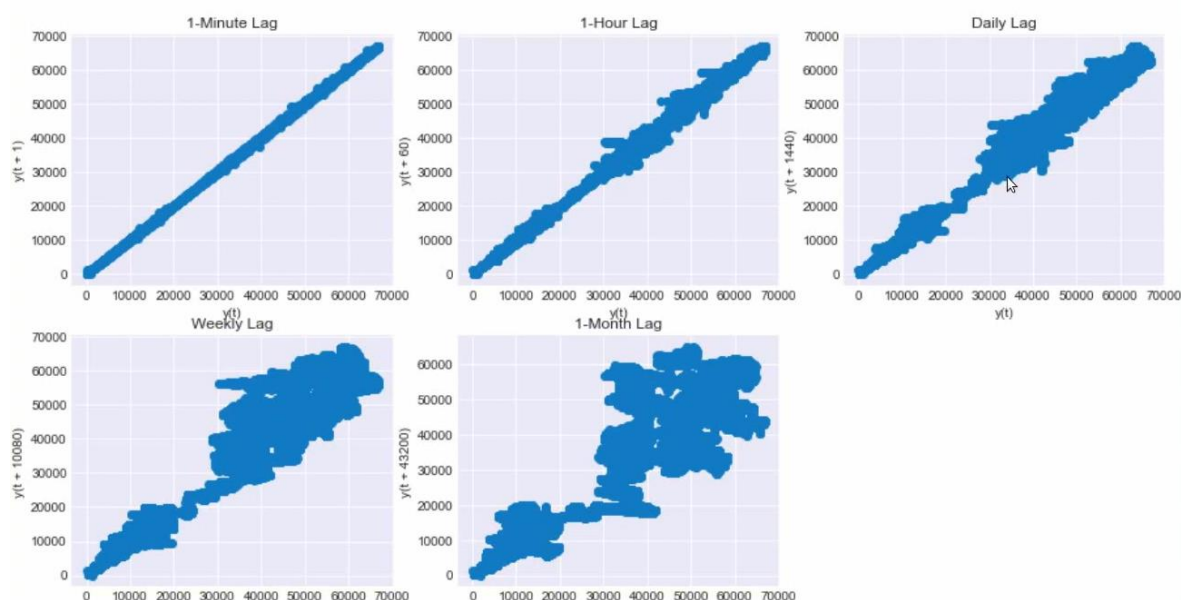


Рисунок 4.2 – Діаграми лагів (Рисунок виконаний самостійно)

Графік лагу перевіряє, чи є набір даних або часовий ряд випадковим чи ні. Випадкові дані не повинні демонструвати будь-яку ідентифіковану структуру на графіку. Невипадкова структура на графіку відставання вказує на те, що базові дані не є випадковими.

З побудованих діаграм можна побачити, що використовувати тижневий та місячний таймфрейми не є доцільним, тому що ідентифікувати структуру на даних діаграмах досить важко, а це означає що кореляція даних низька. На всіх інших розглянутих таймфреймах легко ідентифікувати структуру, тобто кореляція даних є досить високою. Оскільки на хвилинном та годинному таймфреймах буде дуже великий об'єм даних, що може погано вплинути на час навчання моделі та прогнозування, тому під час експерименту будемо використовувати таймфрейм за добу.

Для проведення дослідження та прогнозування ціни закриття криптовалюти на поточний день будемо використовувати два датасети. Перший з яких буде містити ціну відкриття, найвищу та найнижчу ціни та об'єм. Приклад першого датасету наведено нижче (див. рис. 4.3).

	Close	Open	High	Low	Volume
<b>Date</b>					
2020-03-19	0.165037	0.143159	0.170475	0.143130	2590533670
2020-03-20	0.156868	0.165092	0.173812	0.149187	2946481356
2020-03-21	0.158179	0.156715	0.160544	0.153647	2068099167
2020-03-22	0.150214	0.158121	0.161106	0.149621	1864979798
2020-03-23	0.157419	0.149929	0.157933	0.149190	2117431211
...	...	...	...	...	...
2023-03-14	0.374279	0.373790	0.387479	0.367318	1527899940
2023-03-15	0.360558	0.374326	0.377221	0.358760	1062482118
2023-03-16	0.366025	0.360636	0.367265	0.358537	692737535
2023-03-17	0.380382	0.366015	0.381317	0.363593	1046497903
2023-03-18	0.374433	0.380410	0.387869	0.373449	867351027

Рисунок 4.3 – Приклад даних першого датасету (Рисунок виконаний самостійно)

Для прогнозування ціни закриття криптовалюти на поточний день другий датасет буде заснований на показниках технічних індикаторів [23]:

– RSI (Relative Strength Index) – це індикатор імпульсу, який використовується в технічному аналізі. RSI розраховує відношення середнього прибутку до середньої втрати протягом визначеного періоду, зазвичай 14 днів. Значення RSI коливаються від 0 до 100, з показниками понад 70 вважаються перекупленими, а показники нижче 30 вважаються перепроданими.

– MACD (Moving Average Convergence/Divergence) – це популярний технічний індикатор, який використовується для вимірювання зміни ціни цінних паперів, а також для визначення потенційної тенденції, змін у силі, напрямку, імпульсі та тривалості тренду в ціні.

– SMA (Simple Moving Average) – це один з найпростіших технічних індикаторів. SMA розраховується шляхом обчислення середньої ціни закриття за певний період часу. SMA допомагає згладжувати цінову динаміку та показує загальну тенденцію цін на ринку.

– WMA (Weighted Moving Average) – схожий на SMA, але відрізняється в способі розрахунку. В WMA більший ваговий коефіцієнт надається останнім цінам закриття, що дозволяє швидше реагувати на зміну ціни на ринку. Можна використовувати для показування тенденції цін на ринку, а також для виявлення пунктів входу та виходу з позиції.

– EMA (Exponential Moving Average) – також зважені щодо останніх цін, але швидкість зниження між однією ціною та її попередньою ціною не є постійною. Різниця у зменшенні експоненціальна.

Приклад даних другого датасету наведено нижче (див. рис. 4.4)

Date	Close	SMA_5	SMA_10	SMA_20	SMA_30	SMA_50	WMA_10	WMA_20	WMA_30	RSI_14	EMA_10	MACD_8_21_9	MACDh_8_21_9
2020-05-07	0.218917	0.217793	0.218781	0.205230	0.200134	0.188707	0.218403	0.211961	0.206252	62.684797	0.214872	0.009270	-0.000635
2020-05-08	0.219827	0.217876	0.219215	0.206475	0.200766	0.189803	0.218594	0.213352	0.207523	63.241638	0.215773	0.008903	-0.000801
2020-05-09	0.216902	0.217464	0.218162	0.207777	0.201387	0.191004	0.218173	0.214345	0.208564	60.135337	0.215978	0.008080	-0.001299
2020-05-10	0.198176	0.213806	0.216703	0.208511	0.201745	0.191803	0.214539	0.213430	0.208357	44.922523	0.212741	0.004876	-0.003603
2020-05-11	0.194064	0.209577	0.214185	0.208998	0.201945	0.192680	0.210423	0.212054	0.207861	42.386799	0.209345	0.001972	-0.005206
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2023-03-14	0.374279	0.371767	0.373837	0.376565	0.381139	0.389412	0.373675	0.374913	0.377733	46.580437	0.374417	-0.003762	0.000511
2023-03-15	0.360558	0.369755	0.373173	0.375137	0.380768	0.388469	0.371261	0.373389	0.376405	39.154476	0.371897	-0.005176	-0.000723
2023-03-16	0.366025	0.369593	0.372729	0.374514	0.380222	0.387436	0.369961	0.372521	0.375454	43.050211	0.370829	-0.005354	-0.000720
2023-03-17	0.380382	0.371006	0.372712	0.374629	0.379523	0.386841	0.371353	0.373080	0.375464	51.781425	0.372566	-0.003486	0.000918
2023-03-18	0.374433	0.371135	0.371237	0.374449	0.379166	0.386074	0.371666	0.373061	0.375136	48.465680	0.372906	-0.002876	0.001222

Рисунок 4.4 – Приклад даних другого датасету (Рисунок виконаний самостійно)

Аналізуючи побудовані датасети, можна зробити висновок, що вони можуть містити лише дані до вчорашнього дня та тимчасові дані поточного дня. Це також є однією з причин, чому було вирішено зосередитися на прогнозуванні ціни закриття поточного дня, оскільки неможливо отримати ці дані на майбутній період часу. Це і є обмеженням обраних моделей.

### 4.3 Нормалізація даних

Нормалізація даних – це важливий етап попередньої обробки в машинному навчанні, який передбачає перетворення числових характеристик набору даних у загальний масштаб. Мета нормалізації даних полягає в тому, щоб жодна функція не домінувала або не впливала на процес навчання більше, ніж інші.

Для нормалізації даних застосуємо наступні формули [24]:

$$X_{std} = \frac{X - \min(X)}{\max(X) - \min(X)}, \quad (4.1)$$

$$X_{scaled} = X_{std} \cdot (\max - \min) + \min, \quad (4.2)$$

де  $X$  – початкове значення,

$X_{scaled}$  – нормалізоване значення,

$\min(X)$ ,  $\max(X)$  – мінімальне і максимальне значення признака відповідно,

$\min$ ,  $\max$  – бажаний діапазон перетворених даних.

Причини використання нормалізації даних:

– покращена точність – нормалізація допомагає підвищити точність моделей машинного навчання, гарантуючи, що функції мають загальний масштаб, що усуває будь-які зміщення, які можуть виникнути через відмінності у величині функцій;

– надійність – нормалізація робить модель машинного навчання стійкішою до викидів у даних, оскільки це допомагає зменшити вплив екстремальних значень, які можуть спотворити процес навчання;

– інтерпретація – нормалізація полегшує інтерпретацію коефіцієнтів моделі, оскільки коефіцієнти можна безпосередньо порівнювати один з одним, і їх відносну важливість можна легко визначити.

#### 4.4 План-програма експерименту

Для проведення експерименту було обрано стабільне середовище – фізичний пристрій на базі Windows. Технічні характеристики даного пристрою наведені у таблиці 4.4.

Таблиця 4.4 – Характеристики тестового пристрою (Таблиця виконана самостійно)

Характеристика	Значення
Name	HP
CPU	Intel(R) Core(TM) i7-11800H
RAM	32 GB
OS	Windows
SSD	512 GB

Для реалізації обраних моделей була обрана мова програмування Python версії 3.8 та середовище розробки Jupyter Notebook версії 6.5.3. Також буде необхідно використовувати наступні бібліотеки:

- tensorflow 2.11.0;
- yfinance 0.2.12;
- matplotlib 3.7.0;
- numpy 1.23.5;
- pandas 1.5.3;
- pandas-ta 0.3.14;
- scikit-learn 1.2.2.

Для проведення експерименту числові дані повинні бути зібрані для всіх метрик кожного методу.

Для порівняння алгоритмів ARMA, ARIMA, LSTM та GRU визначимо наступні найважливіші критерії:

– MAPE (Mean Absolute Percentage Error) – це статистичний показник, що використовується для вимірювання точності прогнозів. Він вказує на середню відсоткову похибку прогнозів порівняно з фактичними даними. MAPE обчислюється як середнє абсолютне значення відсоткової похибки для кожного прогнозування відносно фактичних даних, поділене на середнє фактичне значення.

– RMSE (Root Mean Square Error) – це показник точності прогнозів, який використовується для вимірювання середньоквадратичної похибки між прогнозованими та фактичними значеннями. RMSE обчислюється шляхом взяття квадратного кореня з середнього значення квадратів відхилень між прогнозом і фактичним значенням. Якщо RMSE менший, то прогнозування більш точне.

– Час тренування моделі.

– Час прогнозування.

## 5 РЕАЛІЗАЦІЯ МОДЕЛЕЙ

Для формування датасетів будемо використовувати бібліотеку `yfinance` для отримання даних з Yahoo Finance. Скориставшись методом `download` можна отримати показники криптовалюти для заданого діапазону часу.

```
today = date.today()
data_end_date = today.strftime("%Y-%m-%d")
start_date = date.today() - timedelta(days=1095)
data_start_date = start_date.strftime("%Y-%m-%d")
df_data = yf.download('XRP-USD', start=data_start_date, end=data_end_date)
```

Для знаходження показників обраних технічних індикаторів було вирішено використовувати бібліотеку `pandas_ta`. Наведемо програмну реалізацію функції додавання показників технічних індикаторів до існуючого датасету.

```
def prepare_ti_dataset(df_data):
    df_data.drop(columns=['Open', 'High', 'Low', 'Volume'], inplace=True)
    CustomStrategy = ta.Strategy(
        name="Technical indicators",
        description="SMA 5, 10, 20, 30, 50, WMA 10, 20, 30, RSI, EMA, MACD",
        ta=[
            {"kind": "sma", "length": 5},
            {"kind": "sma", "length": 10},
            {"kind": "sma", "length": 20},
            {"kind": "sma", "length": 30},
            {"kind": "sma", "length": 50},
            {"kind": "wma", "length": 10},
            {"kind": "wma", "length": 20},
            {"kind": "wma", "length": 30},
            {"kind": "rsi"},
            {"kind": "ema"},
            {"kind": "macd", "fast": 8, "slow": 21},
        ]
    )
```

```
df_data.ta.strategy(CustomStrategy)
df_data = df_data.dropna()
return df_data
```

Наступним кроком була нормалізація даних для перетворення числових характеристик набору даних у загальний масштаб. Код наведено нижче.

```
scaler_train = MinMaxScaler(feature_range=(0, 1))
scaled_train = scaler_train.fit_transform(df_train)
```

Нормалізація даних була імплементована за допомогою класу `MinMaxScaler` з бібліотеки `sklearn`.

### 5.1 Реалізація LSTM та GRU

Для навчання моделей рекурентних нейронних мереж дані часового ряду необхідно підготувати. Дані моделі вимагають надання даних у вигляді набору зразків, де кожен зразок має вхідний компонент  $X$  і вихідний компонент  $Y$ . Модель дізнається, як зіставляти вхідні дані з вихідними на основі наданих прикладів [25]. Будемо використовувати спостереження за попередній момент часу як вхідні дані, а спостереження за поточний момент часу як вихідні дані. Наприклад, маємо наступні дані:

$$[1, 2, 3, 4, 5, 6, 7, 8, 9, 10] \quad (5.1)$$

Наведену вище 10-крокову однофакторну послідовність можна виразити так, що вхідні дані – це 3 часові кроки, а вихідні дані – 1 часовий крок.

Результат перетворень наведено нижче:

$$\begin{array}{cc}
 X & Y \\
 [1, 2, 3] & [4] \\
 [2, 3, 4] & [5] \\
 [3, 4, 5] & [6] \\
 \dots & \dots
 \end{array} \tag{5.2}$$

Програмну реалізацію функції підготовки даних для навчання наведено нижче. При проведенні експерименту значення `look_back` параметру буде задано, як 30.

```
def dataset_generator(dataset, look_back=5):
    dataX, dataY = [], []
    for i in range(len(dataset) - look_back):
        window_size_x = dataset[i:(i + look_back), 1:]
        dataX.append(window_size_x)
        dataY.append(dataset[i + look_back, 0])
    return np.array(dataX), np.array(dataY)
```

Після цього можна перейти до побудови LSTM моделі, використовуючи бібліотеку `tensorflow`. Було вирішено побудувати LSTM модель з двома прихованими шарами. Після кожного шару додається метод регуляризації для того, щоб зменшити перенавчання та покращити продуктивність моделі. Також необхідно додати вихідний шар з алгоритмом оптимізації Adam для навчання мережі та функції втрат, яка використовується для оцінки мережі, мінімізованої за допомогою алгоритму оптимізації. Функція побудови LSTM моделі наведено нижче.

```
def get_lstm_model(timesteps_count, features_count):
    regressor = Sequential()
    regressor.add(LSTM(units = 128, activation =
'relu', return_sequences=True, input_shape = (timesteps_count,
features_count)))
    regressor.add(Dropout(0.2))

    # Adding a second LSTM layer and some Dropout regularisation
```

```

regressor.add(LSTM(units = 64, input_shape = (timesteps_count,
features_count)))
regressor.add(Dropout(0.2))

# Adding the output layer
regressor.add(Dense(units = 1))
regressor.summary()
regressor.compile(optimizer = 'adam', loss = 'mean_squared_error')
return regressor

```

Наведемо програмну реалізацію функції побудови моделі GRU:

```

def get_gru_model(timesteps_count, features_count):
    regressor=Sequential()
    regressor.add(GRU(32,return_sequences=True,input_shape=(
timesteps_count, features_count)))
    regressor.add(GRU(32,return_sequences=True))
    regressor.add(GRU(32))
    regressor.add(Dropout(0.2))
    regressor.add(Dense(1))

    regressor.compile(loss='mean_squared_error',optimizer='adam')
    regressor.summary()

    return regressor

```

На відміну від LSTM було використано три приховані шари. Метод регуляризації, алгоритм оптимізації та функцію втрат було використано такі самі, як і в LSTM.

## 5.2 Реалізація ARMA та ARIMA

Перед побудовою моделей сімейства авторегресії необхідно розділити наші тренувальну та тестову вибірки на компоненти X та Y. Компонента Y у даному випадку це ендogenous параметри моделі, а саме ціна закриття криптовалюти.

Компонента  $X$  – це екзогенні показники, що безпосередньо впливають на ціну закриття криптовалюти, тобто усі інші параметри нашого датасету [26].

Для побудови моделі використовувалась бібліотека `statsmodels`. Програмна реалізація функції побудови моделі ARIMA наведено нижче.

```
model = ARIMA(endog=trainY, exog=trainX, order=[30,1,1], freq='D')
```

Для реалізації моделі ARMA необхідно лише замінити значення параметру  $d$  на нуль. Параметр  $d$  представляє кількість разів, коли часовий ряд розрізнявся. Тобто код для моделі ARMA має вигляд:

```
model = ARIMA(endog=trainY, exog=trainX, order=[1,0,1], freq='D')
```

Для розрахунку показників RMSE та MAPE використовувалась бібліотека `sklearn`.

## 6 ПРОВЕДЕННЯ ЕКСПЕРИМЕНТУ

### 6.1 Модель LSTM

Оберемо криптовалюту XRP для прогнозування за допомогою моделі LSTM. Запустимо нашу модель для першого датасету та перш за все побудуємо графік значень loss функції (див. рис. 6.1) для того, щоб оцінити якість навчання моделі.

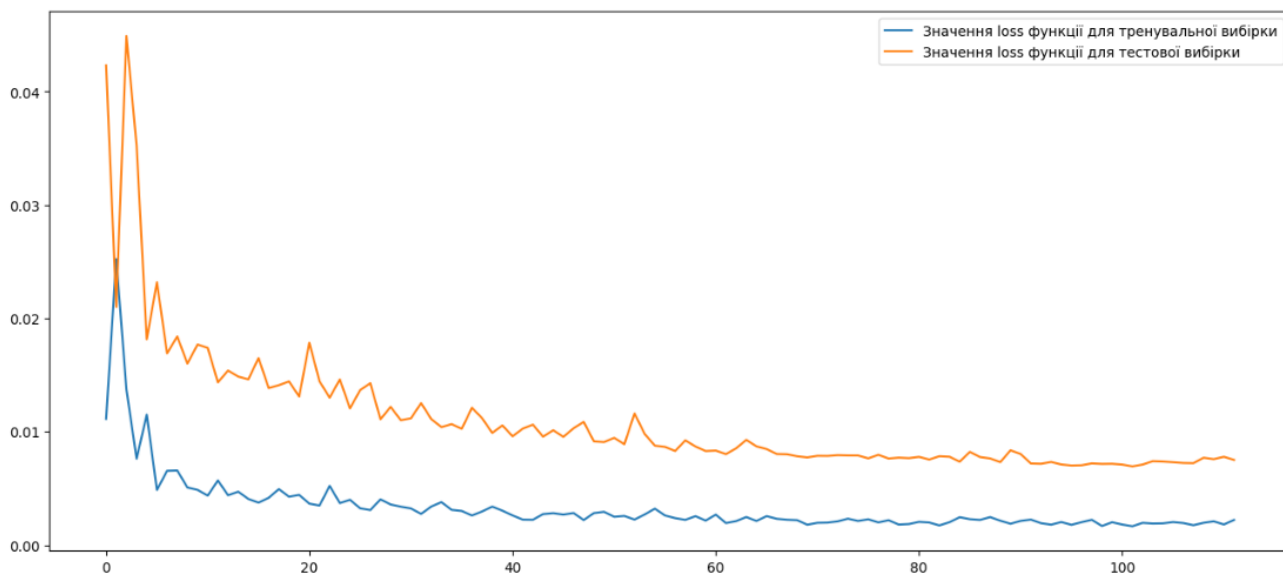


Рисунок 6.1 – Графік loss функції моделі LSTM для першого датасету (Рисунок виконаний самостійно)

Найкраще значення loss функції для тренувального датасету становить 0.0017, для тестового – 0.0069. Аналізуючи графік, можна побачити, що значення функції для тренувального та тестового датасетів зменшуються до певної точки стабільності та мають невеликий розрив [27]. Виходячи з цього, можна зробити висновок, що модель навчена добре.

Побудуємо графік прогнозованої ціни закриття криптовалюти XRP та порівняємо з даними з тестового датасету (див. рис. 6.2). Прогнозована ціна зображена червоним кольором на графіку, справжня – синім.

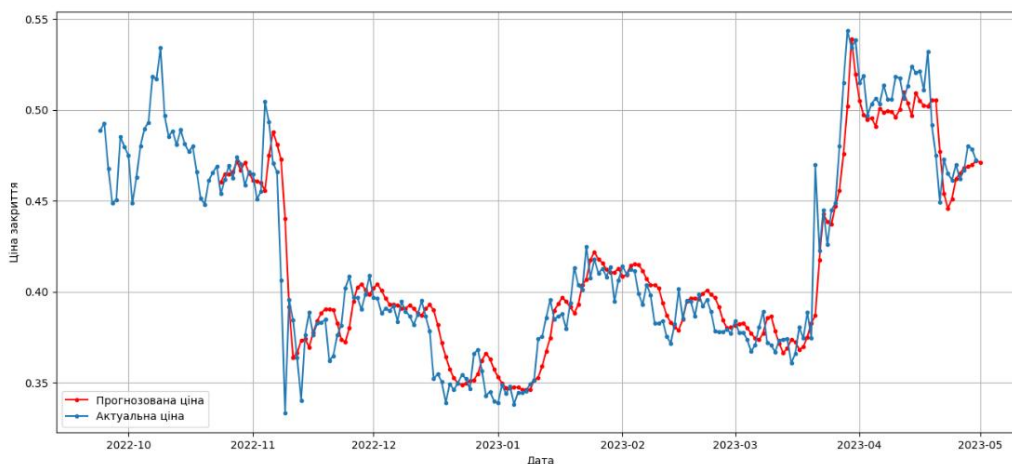


Рисунок 6.2 – Графік прогнозу моделі LSTM для першого датасету (Рисунок виконаний самостійно)

Навчання моделі LSTM для першого датасету зайняло 115.3 секунд, час прогнозування дорівнює 1.12 секундам. Значення статистичних показників наступні: MAPE – 2.84% та RMSE – 0.018.

Далі перейдемо до навчання моделі LSTM, використовуючи другий датасет. Після цього отримали наступний графік значень loss функції (див. рис. 6.3).

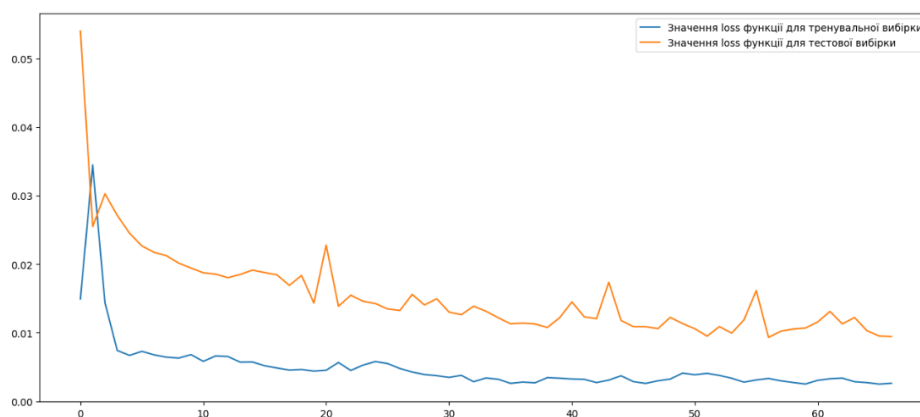


Рисунок 6.3 – Графік loss функції моделі LSTM для другого датасету (Рисунок виконаний самостійно)

Найкраще значення loss функції для тренувального датасету становить 0.0033, для тестового – 0.0093. Аналогічно з попереднім датасетом, значення функції для тренувального та тестового датасетів зменшуються до певної точки стабільності та мають невеликий розрив. Отже, модель навчена добре.

Графік прогнозованої та справжніх цін криптовалюти XRP наведено нижче (див. рис. 6.4).

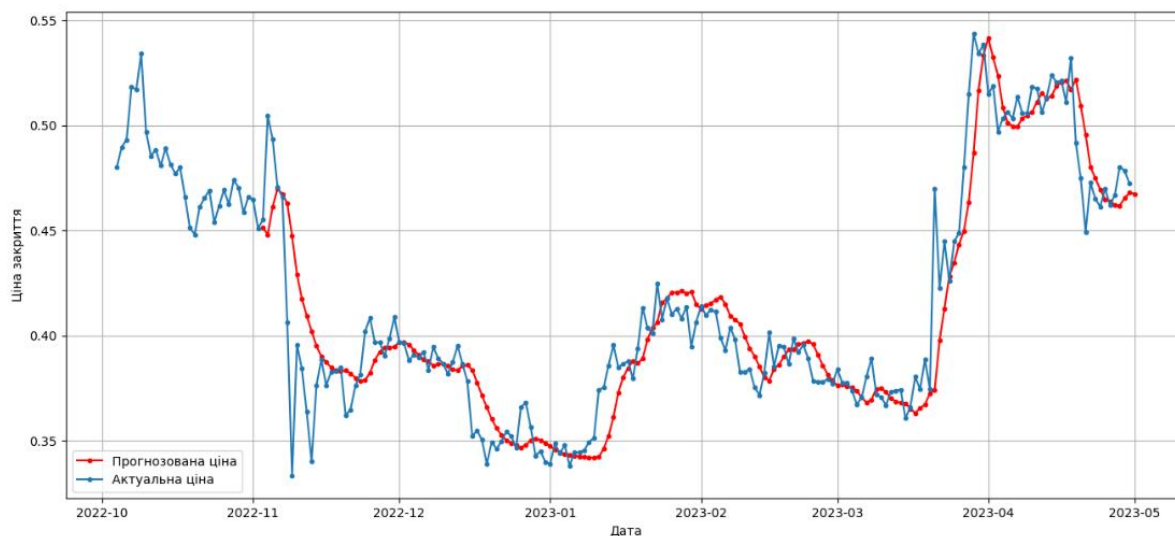


Рисунок 6.4 – Графік прогнозу моделі LSTM для другого датасету (Рисунок виконаний самостійно)

Навчання моделі LSTM для другого датасету зайняло 83.46 секунд, час прогнозування дорівнює 0.92 секундам. Значення статистичних показників наступні: MAPE – 3.14% та RMSE – 0.02.

Для криптовалют Dogecoin та Cardano був також проведений експеримент, використовуючи два датасети. Усі результати були зібрані та наведені нижче (див. табл. 6.1). Параметр час навчання у таблиці позначено, як ЧН, час прогнозу – ЧП.

Таблиця 6.1 – Результати моделі LSTM (Таблиця виконана самостійно)

Токен	Перший датасет (базові показники)				Другий датасет (технічні індикатори)			
	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.
XRP	2.84	0.018	115.3	1.12	3.14	0.02	83.46	0.92
Dogecoin	5.2	0.007	59.34	0.97	6.64	0.008	47	1
Cardano	3.98	0.019	81.73	0.98	4.75	0.021	77.17	0.93

## 6.2 Модель GRU

Для прогнозування за допомогою моделі GRU, оберемо криптовалюту Dogecoin. Запустимо нашу модель для першого датасету та побудуємо графік значень loss функції (див. рис. 6.5) для того, щоб оцінити якість навчання моделі.

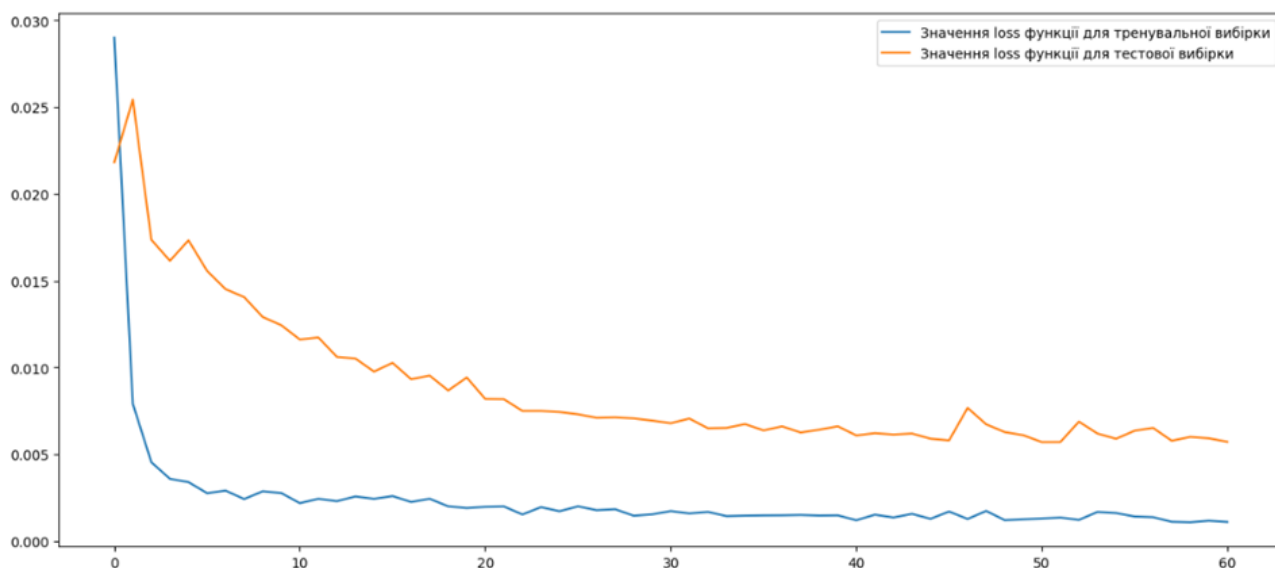


Рисунок 6.5 – Графік loss функції моделі GRU для першого датасету (Рисунок виконаний самостійно)

Найкраще значення loss функції для тренувального датасету становить 0.0013, для тестового – 0.0057. Значення функції для тренувального та тестового датасетів зменшуються до певної точки стабільності, після цього залишаються на певному рівні та мають невеликий розрив. Це свідчить про те, що модель навчена добре.

Графік прогнозованої та справжніх цін криптовалюти Dogecoin наведено нижче (див. рис. 6.6).

Навчання моделі GRU для першого датасету зайняло 86.26 секунд, час прогнозування дорівнює 1.19 секундам. Значення статистичних показників наступні: MAPE – 4.52% та RMSE – 0.006.

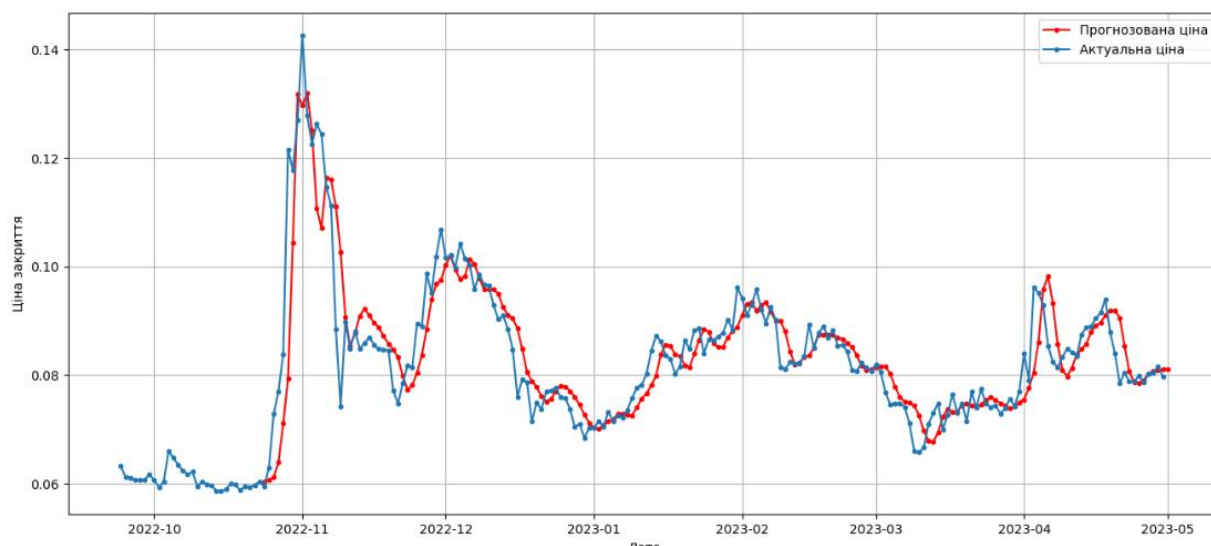


Рисунок 6.6 – Графік прогнозу моделі GRU для першого датасету (Рисунок виконаний самостійно)

Далі перейдемо до навчання моделі GRU, використовуючи другий датасет. Після цього отримали наступний графік значень loss функції (див. рис. 6.7).

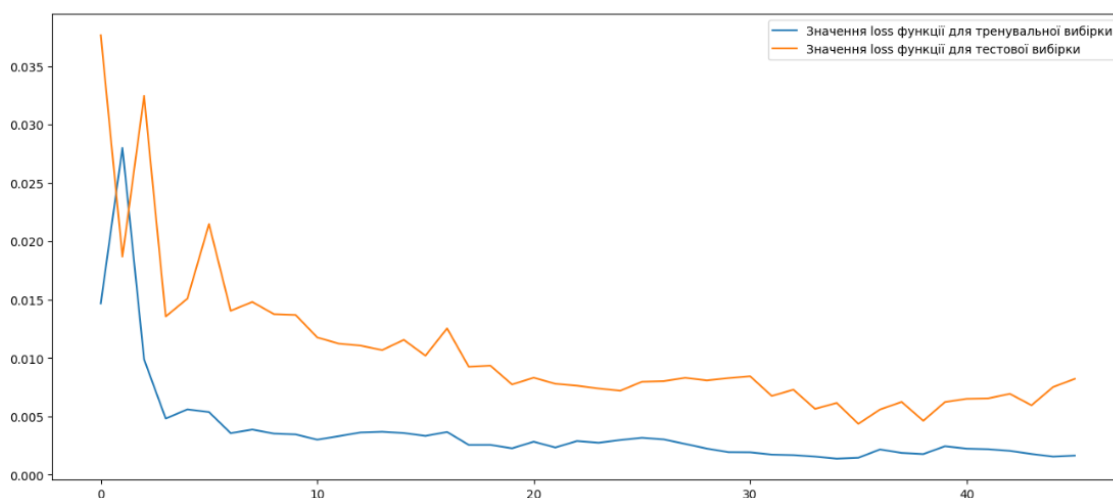


Рисунок 6.7 – Графік loss функції моделі GRU для другого датасету (Рисунок виконаний самостійно)

Найкраще значення loss функції для тренувального датасету становить 0.0015, для тестового – 0.0044. Розрив значень loss функції для тренувального та тестового датасетів є меншим, ніж при навчанні першого датасету. Явища перенавчання не спостерігається. Можемо вважати, що модель навчена добре.

Графік прогнозованої та справжніх цін криптовалюти Dogecoin для другого датасету наведено нижче (див. рис. 6.8).

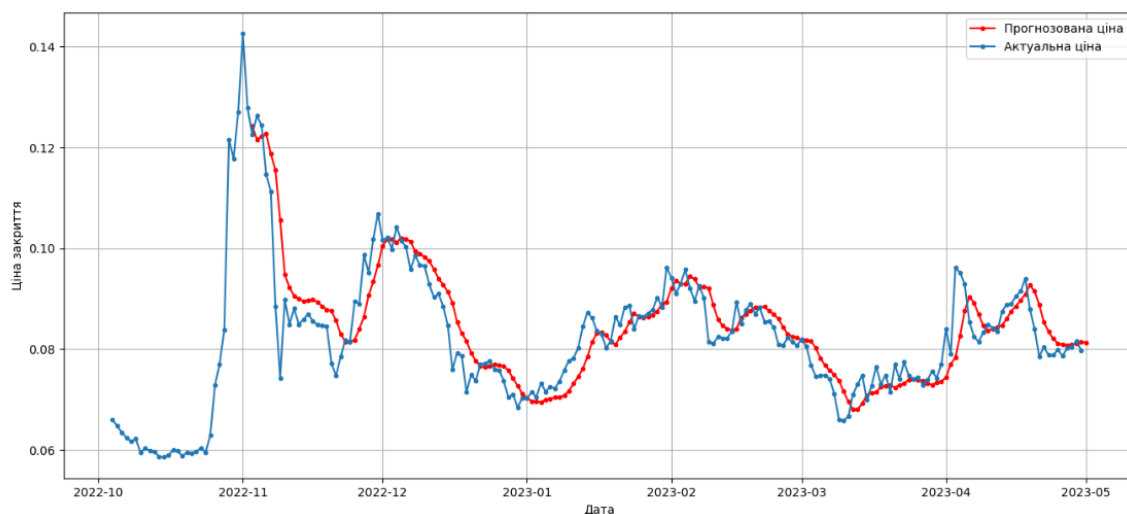


Рисунок 6.8 – Графік прогнозу моделі GRU для другого датасету (Рисунок виконаний самостійно)

Навчання моделі GRU для другого датасету зайняло 90.49 секунд, час прогнозування дорівнює 1.17 секундам. Значення статистичних показників наступні: MAPE – 4.57% та RMSE – 0.006.

Для криптовалют XRP та Cardano був також проведений експеримент, використовуючи два датасети. Усі результати були зібрані та наведені нижче (див. табл. 6.2).

Таблиця 6.2 – Результати моделі GRU (Таблиця виконана самостійно)

Токен	Перший датасет (базові показники)				Другий датасет (технічні індикатори)			
	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.
XRP	2.71	0.017	84.22	1.14	2.83	0.019	102.91	1.18
Dogecoin	4.52	0.006	86.26	1.19	4.57	0.006	90.49	1.17
Cardano	4.13	0.019	44.57	1.19	4.12	0.018	71.9	1.29

### 6.3 Модель ARMA

Для експерименту з моделлю ARMA оберемо криптовалюту Cardano. Запустимо нашу модель для першого датасету. Для визначення якості моделі скористаємось інформаційними критеріями AIC та BIC. Після побудови моделі критерій AIC дорівнює  $-3887.627$ , а BIC становить  $-3710.896$ . Чим нижче показники інформаційних критеріїв, тим краще навчена модель. Можемо зробити висновок, що модель добре навчена для першого датасету.

Графік прогнозованої та справжніх цін криптовалюти Cardano для першого датасету моделі ARMA наведено нижче (див. рис. 6.9).

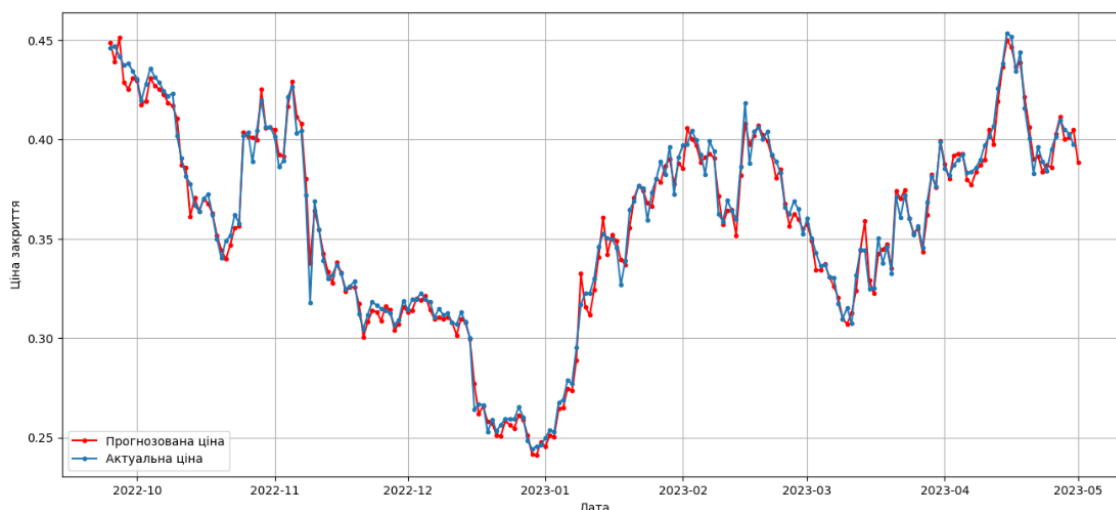


Рисунок 6.9 – Графік прогнозу моделі ARMA для першого датасету (Рисунок виконаний самостійно)

Навчання моделі ARMA для першого датасету зайняло 13.99 секунд, час прогнозування дорівнює 0.19 секунди. Значення статистичних показників наступні: MAPE – 1.21% та RMSE – 0.005.

Перейдемо до навчання моделі ARMA, використовуючи другий датасет. Після побудови моделі критерій AIC дорівнює  $-4091.098$ , а критерій BIC становить  $-3873.471$ . Значення критеріїв дуже близькі до значень моделі, навченої на першому датасеті, отже, модель добре навчена.

Графік прогнозованої та справжніх цін криптовалюти Cardano для другого датасету моделі ARMA наведено нижче (див. рис. 6.10).

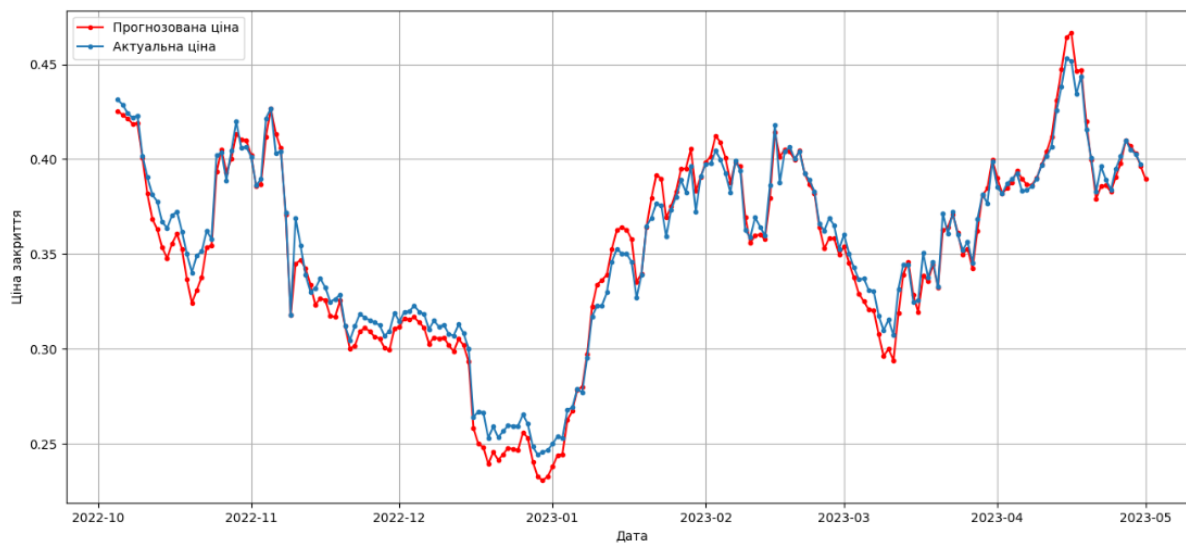


Рисунок 6.10 – Графік прогнозу моделі ARMA для другого датасету (Рисунок виконаний самостійно)

Навчання моделі ARMA для другого датасету зайняло 28.51 секунд, час прогнозування дорівнює 0.24 секунди. Значення статистичних показників наступні: MAPE – 1.98% та RMSE – 0.008.

Для криптовалют XRP та Dogecoin був також проведений експеримент, використовуючи два датасети. Усі результати були зібрані та наведені нижче (див. табл. 6.3).

Таблиця 6.3 – Результати моделі ARMA (Таблиця виконана самостійно)

Токен	Перший датасет (базові показники)				Другий датасет (технічні індикатори)			
	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.
XRP	1.09	0.006	14.1	0.24	0.93	0.005	45.37	0.23
Dogecoin	1.49	0.002	14.11	0.21	1.33	0.002	37.21	0.22
Cardano	1.21	0.005	13.99	0.19	1.98	0.008	28.51	0.24

## 6.4 Модель ARIMA

Для прогнозування за допомогою моделі ARIMA, оберемо криптовалюту Cardano. Запустимо нашу модель для першого датасету. Для визначення якості моделі також скористаємось інформаційними критеріями AIC та BIC. Після побудови моделі критерій AIC дорівнює  $-3519.188$ , а BIC становить  $-3347.274$ . Значення критеріїв нижче, чим у моделі ARMA, що свідчить про те, що модель навчена не так добре.

Графік прогнозованої та справжніх цін криптовалюти Cardano для першого датасету моделі ARIMA наведено нижче (див. рис. 6.11).

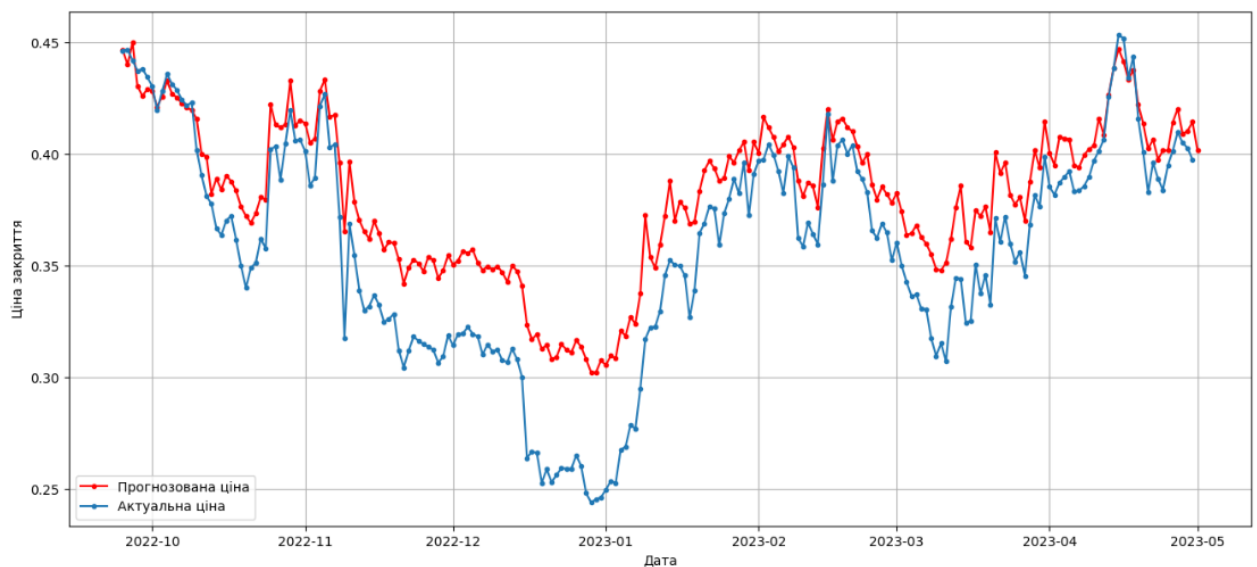


Рисунок 6.11 – Графік прогнозу моделі ARIMA для першого датасету (Рисунок виконаний самостійно)

Навчання моделі ARIMA для першого датасету зайняло 14.01 секунд, час прогнозування дорівнює 0.25 секунди. Значення статистичних показників наступні: MAPE – 9.63% та RMSE – 0.029.

Перейдемо до навчання моделі ARIMA, використовуючи другий датасет. Після побудови моделі критерій AIC дорівнює  $-3503.749$ , а критерій BIC становить  $-3590.907$ , що показує, що модель також навчена гірше, ніж ARMA.

Графік прогнозованої та справжніх цін криптовалюти Cardano для другого датасету моделі ARIMA наведено нижче (див. рис. 6.12).

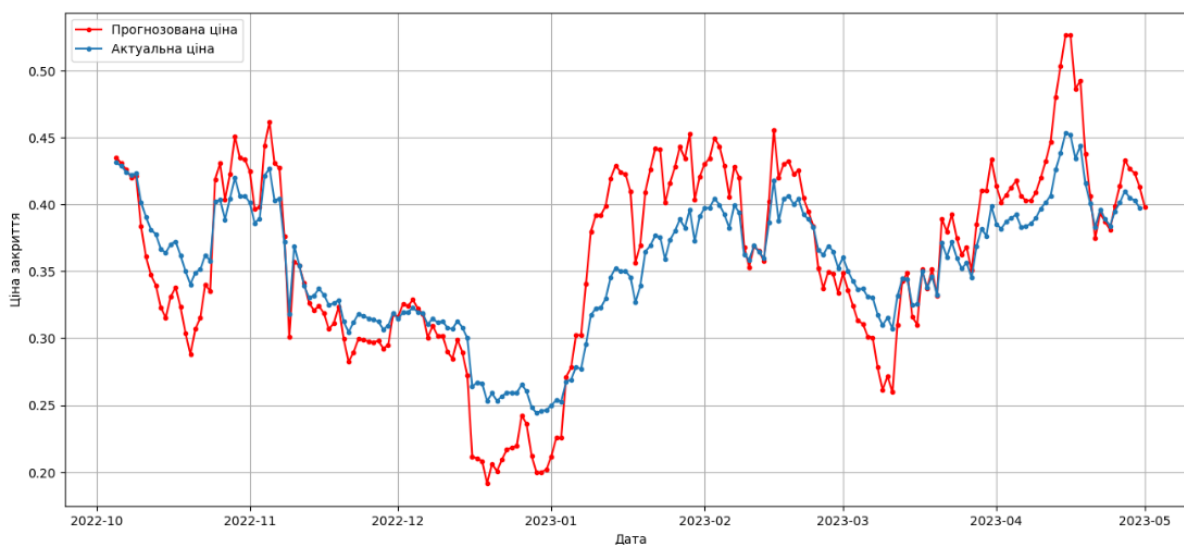


Рисунок 6.12 – Графік прогнозу моделі ARIMA для другого датасету (Рисунок виконаний самостійно)

Навчання моделі ARIMA для другого датасету зайняло 50.89 секунд, час прогнозування дорівнює 0.24 секунди. Значення статистичних показників наступні: MAPE – 7.69% та RMSE – 0.033.

Для криптовалют XRP та Dogecoin був також проведений експеримент, використовуючи два датасети. Усі результати були зібрані та наведені нижче (див. табл. 6.4).

Таблиця 6.4 – Результати моделі ARIMA (Таблиця виконана самостійно)

Токен	Перший датасет (базові показники)				Другий датасет (технічні індикатори)			
	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.
XRP	8.08	0.036	14.79	0.23	5.97	0.019	41.67	0.27
Dogecoin	10.65	0.01	14.38	0.24	6.83	0.021	43.15	0.3
Cardano	9.63	0.029	14.01	0.25	7.69	0.033	50.89	0.24

## 6.5 Аналіз результатів

Для визначення якості моделей рекурентних нейронних мереж LSTM та GRU були розглянуті значення loss функцій. Аналізуючи значення даної функції, було виявлено, що вони для тренувального та тестового датасетів зменшуються до певної точки стабільності, після цього залишаються в певному діапазоні та мають невеликий розрив. Дана поведінка характерна для добре навчених моделей [27]. Отже, можна зробити висновок, що кількість та характер даних в датасетах було правильно обрано, реалізовані архітектури нейронних мереж є оптимальними для даної задачі та явище перенавченості вдалось уникнути.

Для визначення якості моделей сімейства авторегресій з екзогенними показниками ARMA та ARIMA використовувались інформаційні критерії AIC та BIC. Значення даних критеріїв були досить низькими для усіх датасетів та моделей й були дуже схожими між собою. Це свідчить про те, що структурні параметри  $p$ ,  $d$ ,  $q$ , що характеризують порядок для відповідних частин моделі – авторегресійної, інтегрованої та рухомого середнього, були обрані правильно.

Перейдемо до аналізу результатів прогнозування. Для цього розрахуємо середнє арифметичне значення усіх критеріїв для кожної моделі та відповідного датасету. Результат наведено у таблиці 6.5.

Таблиця 6.5 – Результати експерименту (Таблиця виконана самостійно)

Модель	Перший датасет (базові показники)				Другий датасет (технічні індикатори)			
	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.
LSTM	4.01	0.015	85.46	1.02	4.84	0.016	69.21	0.95
GRU	3.79	0.014	71.68	1.17	3.84	0.014	88.43	1.21
ARMA	1.26	0.004	14.07	0.21	1.41	0.005	37.03	0.23
ARIMA	9.45	0.025	14.39	0.24	6.83	0.024	45.24	0.27

Оскільки, MAPE показує середню відсоткову похибку прогнозів порівняно з фактичними даними, тоді розрахуємо точність моделі віднявши від 100% даний показник. Результат наведено у таблиці 6.6.

Таблиця 6.6 – Точність моделей (Таблиця виконана самостійно)

Модель	Точність, %	
	Перший датасет (базові показники)	Другий датасет (технічні індикатори)
LSTM	95.99	95.16
GRU	96.21	96.16
ARMA	98.74	98.59
ARIMA	90.55	93.17

Можемо побачити, що модель ARMA показала найкращі результати по всім критеріям. Це можна пояснити тим, що моделі нейронних мереж LSTM і GRU зазвичай потребують більшого обсягу даних для досягнення кращих результатів, ніж моделі, які базуються на статистичних методах, таких як ARMA [28]. Модель ARMA може показати кращі результати, оскільки вона може краще узагальнювати дані. Тобто для моделей нейронних мереж треба брати датасети більшого об'єму. Якість даних також може впливати на ефективність моделей. Якщо дані мають багато шуму або випадкових змін, нейронні мережі можуть бути менш точними.

Точність моделей LSTM та GRU виявились приблизно однаковими, але все ж таки GRU показала кращі результати. Скоріш за все на це вплинуло те, що GRU має менше параметрів, ніж LSTM, оскільки вона має менше фільтрів [29]. Але при збільшенні об'єму датасетів ситуація може змінитися, через те що LSTM краще розпізнає довгострокові залежності [30].

Модель ARIMA показала найгірші результати точності. Ціни на криптовалюту мають складну та непередбачувану динаміку. Оскільки модель ARIMA використовується для прогнозування часових рядів з незначними відхиленнями від тренду та є менш адаптивною до складних динамічних змін,

можна зробити висновок, що використання даної моделі для прогнозування криптовалют не є раціональним.

## 6.6 Оптимізація моделей

Для оптимізації прогнозування моделей LSTM та GRU складемо один датасет з найвпливовіших характеристик розглянутих датасетів під час експерименту. Для оцінки важливості характеристик скористаємось методом permutation feature importance. У даному методі кожна ознака випадковим чином переміщується у валідаційному наборі даних, і модель знову використовується для прогнозування значень на перемішаних даних [31]. Важливість ознаки визначається на основі того, наскільки точність моделі зменшується при перемішуванні цієї ознаки.

Розрахуємо важливість ознак для першого датасету, результат наведено нижче (див. рис. 6.13).

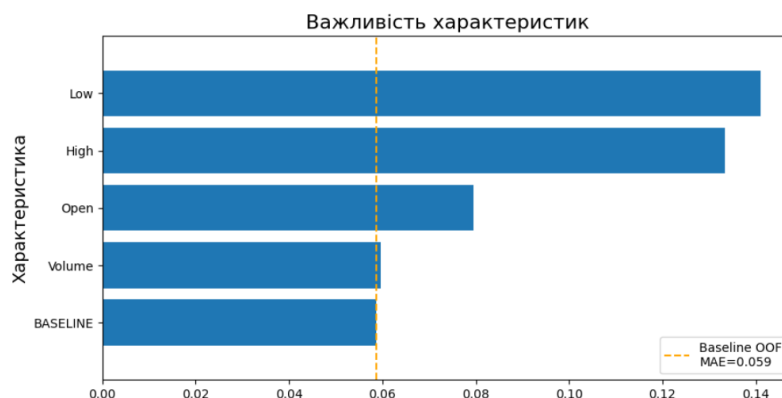


Рисунок 6.13 – Важливість ознак першого датасету для моделей нейронних мереж  
(Рисунок виконаний самостійно)

Розрахуємо важливість ознак для другого датасету, результат наведено нижче (див. рис. 6.14).

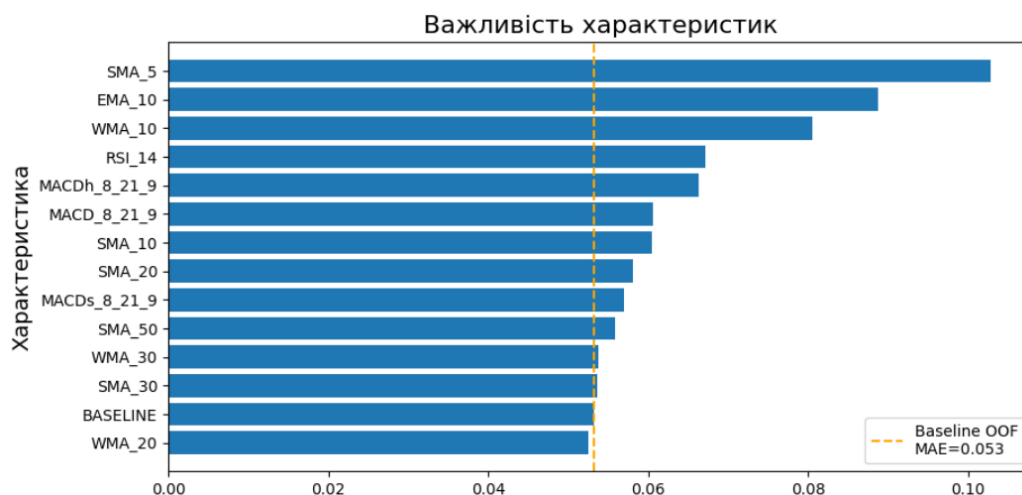


Рисунок 6.14 – Важливість ознак другого датасету для моделей нейронних мереж  
(Рисунок виконаний самостійно)

З отриманих результатів для моделей GRU та LSTM до оптимізованого датасету оберемо наступні ознаки: Low, High, SMA 5, EMA 10 та WMA 10.

Для побудови оптимізованого датасету для моделей ARMA та ARIMA скористаємось отриманими коефіцієнтами після побудови моделей. Приклад наведено нижче (див. рис. 6.15).

```

=====
                                coef
-----
const                0.0019   :
Open                 -0.5172   :
High                 0.8492    :
Low                  0.6615    :
Volume              -3.537e-13  :

```

Рисунок 6.15 – Коефіцієнти ознак першого датасету для моделей сімейства авторегресії (Рисунок виконаний самостійно)

Аналогічним чином було проаналізовано важливість характеристик для другого датасету.

Отже, до оптимізованого датасету для моделей сімейства авторегресії оберемо наступні ознаки: Low, High, WMA 20, MACD, WMA 30.

Проведемо аналогічний експеримент для оптимізованих датасетів. Результати наведено у таблиці 6.7.

Таблиця 6.7 – Результати експерименту для оптимізованих датасетів (Таблиця виконана самостійно)

Модель	MAPE, %	RMSE	ЧН, сек.	ЧП, сек.	Точність, %
LSTM	3.59	0.014	140.08	1.06	96.4
GRU	2.31	0.013	101.6	1.15	97.69
ARMA	1.54	0.006	39.23	0.22	98.46
ARIMA	9.5	0.026	41.63	0.3	90.5

Можемо побачити, що модель ARMA також показала найкращі результати по всім критеріям. Точність залишилася майже на тому ж самому рівні, але збільшився час навчання моделі.

Модель ARIMA показала приблизно ті ж самі результати у порівнянні з попереднім експериментом та залишилась найгіршою у точності прогнозування.

Моделі LSTM та GRU показали кращі результати точності, використовуючи оптимізований датасет, але як і в інших моделях, збільшився час навчання.

Для моделі GRU час навчання збільшився незначно у порівнянні з датасетом технічних індикаторів.

Для моделі LSTM час навчання збільшився майже у два рази у порівнянні з датасетом технічних індикаторів.

Можемо зробити висновок, що в цілому комбінований датасет допоміг збільшити точність деяких моделей, але незначно й при збільшенні часу навчання моделі.

## 6.7 Практична цінність отриманих результатів

Отримані результати дослідження дають можливість порівняти ефективність різних методів прогнозування ціни закриття криптовалюти на поточний день.

Використовуючи дані результати, планується розробити сервіс для прогнозування ціни закриття криптовалюти на поточний день. При підключенні даного сервісу до певної біржі через API, він зможе аналізувати ситуацію на ринку, активні ордери та поради, коли краще входити в ордер чи виходити з вже існуючого. Даний сервіс буде корисним для трейдерів, фокусом яких є ф'ючерсна внутрішньоденна торгівля.

Окрім того, дослідження може бути корисним для академічних дослідників та студентів, які займаються аналізом фінансових ринків та прогнозуванням цін на активи [32]. Результати можуть бути використані для подальшого дослідження та розвитку нових методів прогнозування.

## ВИСНОВКИ

У ході виконання магістерської роботи був проведений аналіз предметної області, існуючих систем прогнозування курсу криптовалют та відповідних наукових джерел. За його результатами було виявлено, що найбільш вживаними алгоритмами прогнозування є алгоритми сімейства авторегресії та рекурентних нейронних мереж, а саме:

- LSTM (Long Short-Term Memory): рекурентна нейронна мережа довгої короткострокової пам'яті;
- GRU (Gated Recurrent Unit): рекурентна нейронна мережа;
- ARMA (Autoregressive Moving Average): модель авторегресії рухомого середнього з екзогенними показниками;
- ARIMA (Autoregressive Integrated Moving Average): модель авторегресії інтегрованого рухомого середнього з екзогенними показниками.

З аналізу наукових джерел було зроблено висновок, що треба зосередитися на короткостроковому прогнозуванні, тому що довгостроковий прогноз криптовалюти має низьку точність через високу волатильність ринку та впливу багатьох чинників. Саме тому було вирішено зосередитися на прогнозуванні ціни закриття криптовалюти для поточного дня.

Було визначено методологію проведення експерименту, план-програму експерименту та критерії порівняння обраних моделей. За допомогою лінійної адитивної згортки з нормуючими множниками обрали криптовалюти для дослідження. Підготували два датасети та визначилися з методом нормалізації даних. За допомогою мови програмування Python були реалізовані обрані моделі.

Після реалізації досліджених моделей було проведено експеримент, з якого можна зробити наступні висновки, щодо прогнозування ціни закриття криптовалюти на поточний день:

- модель ARMA показала найкращі результати по всім критеріям через те, що краще узагальнює дані, навіть при невеликому розміру датасету;

- критерій час прогнозування можна не враховувати при порівнянні, оскільки значення є низьким для усіх моделей та незначно відрізняється між ними;
- модель GRU виявилась більш точною, ніж LSTM, оскільки LSTM потребує набагато більшого розміру датасету для розпізнання довгострокових залежностей;
- за часом навчання модель GRU є також кращею, ніж LSTM, оскільки має менше параметрів та фільтрів, тобто простішу архітектуру;
- модель ARIMA показала найгірші результати точності, оскільки є менш адаптивною до складних динамічних змін;
- модель ARIMA показала кращі результати за критерієм «час навчання», ніж LSTM та GRU, але оскільки точність прогнозу є досить низькою, то використання даної моделі для прогнозування криптовалют не є раціональним;
- згідно зі значень loss функцій та інформаційних критеріїв AIC та BIC – усі моделі були добре навчені.

Для оптимізації прогнозування моделей було підготовлено один датасет з найвпливовіших характеристик розглянутих датасетів під час експерименту та проведено повторний експеримент, який показав, що точність моделей GRU та LSTM трохи збільшилась, точність моделей ARIMA та ARMA змінились незначно, можна сказати, що залишилась на тому ж рівні, але збільшився час навчання усіх моделей.

Отже, можна зробити висновок, що модель ARMA є найкращою для прогнозування ціни закриття криптовалют. Точність моделей GRU та LSTM є також високою, тому їх також можливо використовувати.

Отримані результати дослідження можна використовувати для розробки сервісу прогнозування ціни закриття криптовалют та для подальшого дослідження та розвитку нових методів прогнозування.

Виходячи з вищезазначеного, можемо стверджувати виконання поставленого завдання – проведення дослідження для порівняння ефективності застосування алгоритмів сімейства авторегресії ARMA, ARIMA та алгоритмів рекурентних нейронних мереж LSTM, GRU для прогнозування ціни закриття криптовалюти на поточний день.

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Arvind Narayanan. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction / Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller & Steven Goldfeder // Princeton University Press. 2016 – 304 с.
2. Bariviera A.F., Basgall M.J., Hasperué W., Naiouf M. Some stylized facts of the Bitcoin market. *Physica A: Statistical Mechanics and its Applications*. 2017;484:82–90. DOI: 10.1016/j.physa.2017.04.159.
3. J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll and E. W. Felten, "SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies," 2015 IEEE Symposium on Security and Privacy, 2015, pp. 104-121, doi: 10.1109/SP.2015.14.
4. S. Pillai, D. Biyani, R. Motghare and D. Karia, "Price Prediction and Notification System for cryptocurrency Share Market Trading," 2021 International Conference on Communication information and Computing Technology (ICCICT), 2021, pp. 1-7, doi:10.1109/ICCICT50803.2021.9510122.
5. Crypto Predictions. URL: <https://cryptopredictions.com/> (дата звернення: 25.03.2023).
6. Coincodex. URL: <https://coincodex.com/> (дата звернення: 25.03.2023).
7. Trading Beasts. URL: <https://tradingbeasts.com> (дата звернення: 25.03.2023).
8. Coin Market cap. URL: <https://coinmarketcap.com/> (дата звернення: 25.03.2023).
9. Smelyakov, K., Bizkrovnyi, O., Sharonova, N., Smelyakov, S., Chupryna, A. Building of Regression Models for Cryptocurrency Price Prediction CEUR Workshop Proceedings, 2022, 3171, pp. 1216–1232.
10. Gordiievych, A. , Shubin, I. Forecasting of airfare prices using time series Gordiievych, A. , Shubin, I. 2015 Information Technologies in Innovation Business Conference, ITIB 2015 - Proceedings, 2015, с. 68-71, 7355055.
11. E. Karakoyun and A. Cibikdiken, "Comparison of arima time series model and lstm deep learning algorithm for bitcoin price forecasting," in The 13th Multidisciplinary Academic Conference in Prague, vol. 2018, pp. 171–180, 2018.

12. T. Zoumpikas, E. Houstis, and M. Vavalis, “Eth analysis and predictions utilizing deep learning,” *Expert Systems with Applications*, vol. 162, p. 113866, 2020.

13. Elman J. L. Finding structure in time // *Cognitive science*. – 1990. – Vol. 14, no. 2. – Pp. 179-211.

14. Bouoiyour J., Selmi R. Bitcoin price: Is it really that new round of volatility can be on way? Munich Personal RePE Archive. MPRA Paper. 2015;(6558). URL: [https://mpa.ub.uni-muenchen.de/65580/1/MPRA\\_paper\\_65580.pdf](https://mpa.ub.uni-muenchen.de/65580/1/MPRA_paper_65580.pdf) (дата звернення: 01.04.2023).

15. Avdieiev O.D., Kauk V. I., “Research of methods of forecasting the dynamics of the course of cryptocurrencies”, 2023.

16. Transformer neural network. URL: <https://builtin.com/artificial-intelligence/transformer-neural-network> (дата звернення: 03.04.2023).

17. Hochreiter S., Schmidhuber J. Long short-term memory // *Neural computation*. – 1997. – Vol. 9, no. 8. – Pp. 1735 –1780.

18. Multi-resolution linear prediction based features for audio onset detection with bidirectional LSTM neural networks / E. Marchi [et al.] // *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. – IEEE. 2014. – Pp. 2164-2168.

19. A. Dutta, S. Kumar, and M. Basu, “A gated recurrent unit approach to bitcoin price prediction,” *Journal of Risk and Financial Management*, vol. 13, p. 23, Feb 2020.

20. Understanding The Different Types of Cryptocurrency. URL: <https://www.sofi.com/learn/content/understanding-the-different-types-of-cryptocurrency> (дата звернення: 09.04.2023).

21. Pieters G., Vivanco S. Financial regulations and price inconsistencies across Bitcoin markets. *Information Economics and Policy*. 2017;39:1–14. DOI: 10.1016/j.infoecopol.2017.02.002.

22. Yahoo finance URL: <https://finance.yahoo.com> (дата звернення: 09.04.2023).

23. Most popular indicators. URL: <https://www.lcx.com/crypto-trading-toolkit-5-most-popular-indicators/> (дата звернення: 09.04.2023).

24. Machine learning normalization. URL: <https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/> (дата звернення: 09.04.2023).

25. A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms. URL: <https://www.mdpi.com/2673-2688/2/4/30/pdf> (дата звернення: 11.04.2023).

26. Perron P. Further evidence on breaking trend functions in macroeconomic variables. *Journal of Econometrics*. 1997;80(2):355–385. DOI: 10.1016/S 0304–4076(97)00049–3.

27. How to use Learning Curves to Diagnose Machine Learning Model Performance. URL: <https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/> (дата звернення: 11.04.2023).

28. M. Nabipour, P. Nayyeri, H. Jabani, S. S., and A. Mosavi, “Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis,” *IEEE Access*, vol. 8, pp. 150199–150212, 2020.

29. Feature Importance for Any Model using Permutation. URL: [https://medium.com/@T\\_Jen/feature-importance-for-any-model-using-permutation-7997b7287aa](https://medium.com/@T_Jen/feature-importance-for-any-model-using-permutation-7997b7287aa) (дата звернення: 13.04.2023).

30. Investigation of the deep learning approaches to classify emotions in texts / D. Nazarenko et al. *Proceedings of the 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS 2021)*. Volume I: Main Conference, Lviv, 22–23 April 2021. 2021. P. 206–224.

31. Gruzdo, I., Kyrychenko, I., Tereshchenko, G., Shanidze, N. Metrics applicable for evaluating software at the design stage. 2021 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS-2021), 2021. – CEURWS, 2021, ISSN 16130073. - Volume 2870, PP. 916-936.

32. Ворочек О.Г, Дударь В.В. Исследование интеллектуальных средств анализа и прогнозирования. *Журнал «Вестник» Херсонского национального технического университета №4(27)*, 2007.

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ ЗА НАУКОВИМИ НАПРЯМАМИ  
КЕРІВНИКА ТА НАУКОВЦІВ КАФЕДРИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ**

9. Smelyakov, K., Bizkrovnyi, O., Sharonova, N., Smelyakov, S., Chupryna, A. Building of Regression Models for Cryptocurrency Price Prediction CEUR Workshop Proceedings, 2022, 3171, pp. 1216–1232.

10. Gordiievych, A. , Shubin, I. Forecasting of airfare prices using time series Gordiievych, A. , Shubin, I. 2015 Information Technologies in Innovation Business Conference, ITIB 2015 - Proceedings, 2015, с. 68-71, 7355055.

30. Investigation of the deep learning approaches to classify emotions in texts / D. Nazarenko et al. Proceedings of the 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS 2021). Volume I: Main Conference, Lviv, 22–23 April 2021. 2021. P. 206–224.

31. Gruzdo, I., Kyrychenko, I., Tereshchenko, G., Shanidze, N. Metrics applicable for evaluating software at the design stage. 2021 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS-2021), 2021. – CEURWS, 2021, ISSN 16130073. - Volume 2870, PP. 916-936.

32. Ворочек О.Г, Дударь В.В. Исследование интеллектуальных средств анализа и прогнозирования. Журнал «Вестник» Херсонского национального технического университета №4(27), 2007.