

УДК: 004.8; 338.2

ПРОБЛЕМИ ЗАСТОСУВАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ ПРОГНОЗУВАННЯ ЦІНИ ВІРТУАЛЬНИХ АКТИВІВ

Андрій Цемко¹, Максим Мацьків²

Львівський національний університет імені Івана Франка

¹ Кафедра радіофізики та комп'ютерних технологій,
вул. Ген. Тарнавського, 107, 79017 Львів, Україна

² Кафедра системного проектування,
вул. Драгоманова, 50, 79005 Львів, Україна

Цемко, А.М., Мацьків, М.А. (2025). Проблеми застосування нейронних мереж для прогнозування ціни віртуальних активів. *Електроніка та інформаційні технології*, 29, 69-78.
<https://doi.org/10.30970/eli.29.7>

АНОТАЦІЯ

Вступ. Прогнозування цін віртуальних активів є важливим завданням через їх високу мінливість. Нейронні мережі широко використовуються для таких задач, але часто стикаються з проблемою наївних передбачень, коли наступне значення надто схоже на попереднє, що знижує ефективність прогнозування.

Матеріали та методи. Дослідження проводилося на основі даних динаміки цін Bitcoin (біржа Coinbase) за період 4 травня 2021 – 9 квітня 2024. Розглянуто два методи нормалізації даних цін: лінійну (min-max) та нормалізацію співвідношення (ratio), яка забезпечує стаціонарність даних. Нейронна мережа із двонаправленими LSTM-шарами навчалася на 10 попередніх значеннях цін для прогнозування одного наступного. Використано бібліотеку TensorFlow, зокрема інтерфейс Keras, а процес навчання складався з: 150 навчальних епох, тренувальних вибірок по 32 навчальні пари, оптимізатор Adam і функція втрат MSE.

Результати й обговорення. Лінійна нормалізація показала найгірші результати, оскільки модель втрачає здатність до прогнозування, якщо майбутні значення виходять за межі навчальної вибірки. Натомість нормалізація відношення продемонструвала значно кращі результати, дозволяючи моделі враховувати динаміку змін навіть за межами мінімальних і максимальних значень навчальних даних. Використання додаткового множника для нормалізованих даних ($k = 400$) покращило результати для тренувальних даних, хоча на тестовому наборі покращення було незначним. Додавання додаткових характеристик цін (під час відкриття, закриття торгів та мінімальна ціна протягом цього періоду) не змогло усунути проблему зсуву прогнозів. Аналіз результатів показав, що модель часто базується на повторенні попередніх значень, що свідчить про її обмеженість у захопленні складних взаємозв'язків у даних.

Висновки. Дослідження підтверджує, що нейронні мережі мають обмеження у задачах прогнозування цін віртуальних активів. Проблема наївного передбачення є ключовою перешкодою, яка обмежує ефективність моделей. Використання нормалізації відношення із множником k покращує точність, але цього недостатньо для вирішення основної проблеми. Необхідні нові підходи чи вдосконалення існуючих методів, що дозволять моделі краще розуміти складні взаємозв'язки у часових рядах і надійно прогнозувати майбутні зміни.

Ключові слова: рекурентні нейромережі, прогнозування, віртуальні активи.



© 2025 Andrii Tsemko & Maxym Matskiv. Published by the Ivan Franko National University of Lviv on behalf of Електроніка та інформаційні технології / Electronics and information technologies. This is an Open Access article distributed under the terms of the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.

ВСТУП

Метою роботи є висвітлення проблеми використання нейронних мереж для прогнозування майбутніх значень ціни віртуальних активів базуючись на динаміці їхньої зміни за попередні інтервали часу. Під час дослідження застосування нейронних мереж було виявлено фундаментальну проблему, яка потребує глибшого дослідження і висвітлення в задачах прогнозування часових рядів.

Опираючись на результати минулої статті [1], вирішено провести поглиблене дослідження методів нормалізації даних динаміки цін, які в подальшому будуть використані для навчання нейронних мереж. Проте під час цього дослідження виявлено проблему, коли нейронна мережа передбачає наступне значення, як дуже схоже на попереднє, що давалося на вхід (так зване передбачення “як вчора”) [2].

Провівши глибший аналіз літератури, ми побачили, що невеликий розмір нейронної мережі не є основною проблемою. Дана проблема передбачення часто виникає у схожих підходах з нейронними мережами для задач прогнозування. Опираючись на результати дослідження авторів статті “Are Transformers Effective for Time Series Forecasting?”, складніші структури нейронних мереж такі як трансформери, показують точність прогнозуванням на тому самому рівні як і простіші моделі [3]. Тому аспект складності нейронної мережі для задачі прогнозуваннями нами досліджувався лише поверхнево.

Для ефективного процесу навчання нейронних мереж варто використовувати нормалізовані дані. Одним з найпоширенішим методом нормалізації є метод лінійної нормалізації (min-max) [4]. Зазвичай, в цій нормалізації значення зводяться до проміжку від 0 до 1, де 0 – це мінімальне значення (min), а 1 – це максимальне (max). Формула лінійної нормалізації даних:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}, \quad (1)$$

де X – оригінальні дані;

X_{min} та X_{max} – мінімальне та максимальне значення оригінальних даних відповідно;

X_{norm} – нормалізовані дані.

Незважаючи на результати навчання нейронної мережі, використання лінійної нормалізації має недолік, який потенційно може зменшити ефективність застосування нейронної мережі для прогнозування подальших змін ціни. У випадку, якщо майбутні значення цін вийдуть за межі мінімального і максимального значень даних, використаних у навчанні, нейронна мережа не зможе спрогнозувати подальшу зміну цін, оскільки вона не бачила таких значень. Цю проблему було виявлено за минулого дослідження, що відображено на Рис. 1.

Для вирішення цієї проблеми запропоновано використати інший підхід нормалізації даних – нормалізації відношення (ratio). Її суть полягає в зображенні даних як відношення наступного значення $n+1$ до поточного n (або поточного n до попереднього $n-1$). Такий підхід нормалізації також зустрічається в статтях під назвою відсоткова нормалізація. Дані після застосування такої нормалізації знаходяться у проміжку між 0,85 і 1,15, де значення менше 1 відповідає зниженню ціни, а значення більше 1 – зростанню. Таке представлення даних є стаціонарним з середнім значенням в околі 1, що дозволяє використовувати навчену нейронну мережу для прогнозування майбутніх даних, навіть при виході даних за оригінальні мінімальні і максимальні значення, які були використані під час навчання нейронної мережі (Рис. 2).

Отримані дані можна охарактеризувати як стаціонарний процес, оскільки середнє значення не змінюється з часом. Для підвищення ефективності навчання



Рис. 1. Порівняння прогнозованих значень нейронної мережі з оригінальними. Синя лінія – навчальні дані; помаранчева – тестувальні; зелена – прогнозовані значення.

Fig. 1. Comparison of the predicted values of the neural network with the original ones. Blue line – training data; orange – testing data; green – predicted data.

нейронної мережі, ми використовуємо додатковий множник для нормалізованих даних, суть якого полягає у тому, щоб збільшити середнє квадратичне відхилення. Формула нормалізації відношення має вигляд:

$$X_{norm} = \left(\frac{X_n}{X_{n-1}} - 1 \right) \cdot k, \quad (2)$$

де k – множник нормалізації.

Метою даного дослідження є аналіз ефективності альтернативних підходів нормалізації даних у задачах прогнозування часових рядів на основі нейронних мереж для вирішення виявлених проблем точності та адаптивності моделей.

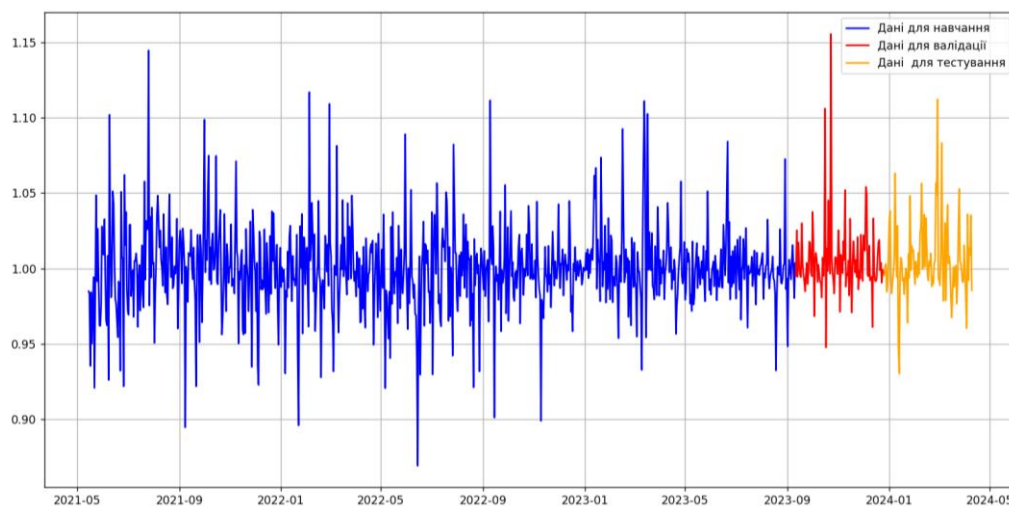


Рис. 2. Дані після нормалізації відношення. Синя лінія – навчальні дані; помаранчева – тестувальні; червона – дані перевірки.

Fig. 2. Ratio normalized data. Blue line – training data; orange – testing data; red – validation data.

МАТЕРІАЛИ ТА МЕТОДИ

Всі описані результати дослідження в цій статті були виконані з використанням даних динаміки цін криптовалюти Bitcoin відносно долару США. Ці дані було взято з біржі Coinbase [5]. Для дослідження було використано дані в часовому проміжку від 4 травня 2021 року до 9 квітня 2024 року, а саме щоденне співвідношення вартості обраних активів (Рис. 3).

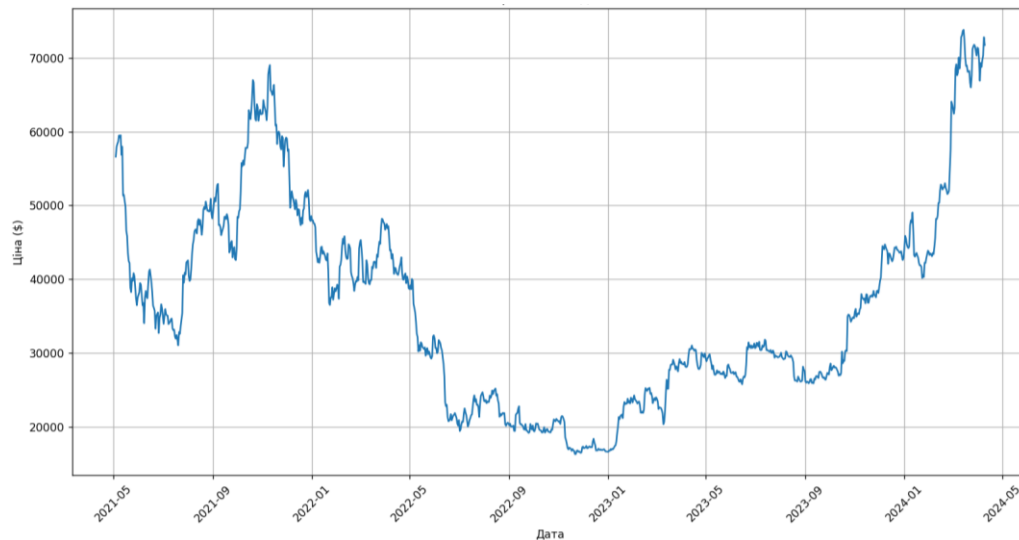


Рис. 3. Максимальна вартість Bitcoin'у по днях з біржі Coinbase
Fig. 3. Maximum daily Bitcoin price from the Coinbase exchange

Для створення та навчання нейронної мережі, було використано бібліотеку TensorFlow [6] мови програмування Python від компанії Google, з відкритим кодом. Бібліотека TensorFlow містить в собі інтерфейс Keras, який дозволяє створювати структури нейронних мереж пошарово, налаштовуючи відповідні параметри кожного з шарів. Так, для створення рекурентних нейронних мереж інтерфейс Keras надає можливість створення таких шарів: RNN (Recurrent Neural Networks, Рекурентна нейронна мережа), LSTM (Long Short-Term Memory, Довга короткочасна пам'ять) та GRU (Gated Recurrent Unit, Вентильний рекурентний вузол) [7, 8].

У моделі нейронної мережі використовується архітектура з двонаправленими LSTM-шарами, що дозволяють враховувати контекст як з попередніх, так і з наступних кроків послідовності. Перший шар містить 50 нейронів передає всю послідовність на наступний шар з функцією активації ReLU (Rectified Linear Unit, Випрямлений лінійний вузол або Випрямляч). Далі йде шар вимкнення 20% нейронів для запобігання перенавантаженню. Другий двонаправлений шар LSTM містить 54 нейрони та вже не повертає послідовність. Потім додаються два щільні (Dense) шари: перший з 50 нейронами і функцією активації ReLU, другий з 25 нейронами без функції активації. Після кожного з них також відбувається вимкнення 20% нейронів для регуляризації. Після останнього вихідного шару нейронна мережа передає згенероване наступне значення (Рис. 4).

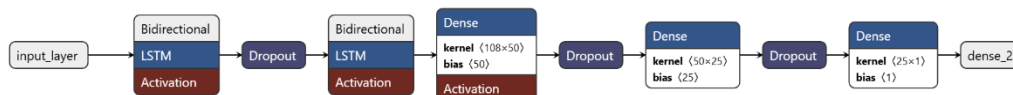


Рис. 4. Структура нейронної мережі.
Fig. 4. Neural network structure.

Для оцінки точності прогнозування моделі використовувався метод середньоквадратичної похибки (Mean Squared Error, MSE). Дані попередньо нормалізувалися за методами, детально описаними у попередньому розділі.

РЕЗУЛЬТАТИ ТА ОБГОВОРЕННЯ

Для дослідження було вирішено навчити нейронну мережу, яка буде генерувати прогноз наступного значення ціни на основі динаміки цін за останні 10 днів. Навчання нейронної мережі проводилось протягом 150 епох, з використання вибірок по 32 набори тренувальних пар. Використано дані за 1072 дні, які було розділено у співвідношенні 80% до 20% на дані для навчання та верифікації. В якості оптимізатора ваг нейронної мережі використано оптимізатор Adam і функцію обрахунку втрат – середньоквадратичну похибку (Mean Squared Error, MSE). Під час процесу навчання для функції втрат використовується значення похибки нормалізованих даних. Результати навчання (наведені у таблиці 1) обраховуються на основі денормалізованих даних.

Таблиця 1. Результати навчання нейронної мережі прогнозування одного значення на основі 10-ти вхідних параметрів лінійно нормалізованих даних.
Table 1. Neural network training results predicting a single value based on 10 input parameters linearly normalized data.

	Навчальні дані	Тестувальні дані
Середньоквадратична похибка (\$)	2,29 млн	7,35 млн

Формула денормалізації лінійно нормалізованих даних має вигляд:

$$X_{denorm} = X_{norm} * (X_{max} - X_{min}) + X_{min}, \quad (3)$$

а формула для денормалізації даних для нормалізації відношення:

$$X_{denorm} = \left(\frac{X_{norm}}{k} + 1 \right) * X_{n-1}. \quad (4)$$

Нами було досліджено вплив значення множника k з формули (2) на результати навчання нейронної мережі. Було досліджено ефективність навчання нейронної мережі при значення множника 1, підбраного значення 36, що відповідає представленню даних стаціонарного процесу характеристикою якого є значення середньоквадратичного відхилення (СКВ) рівне 1 та значення множника 400, що показує кращі результати. Результати середньоквадратичних похибок (MSE) після денормалізації даних занесені до таблиці 2.

Аналізуючи результати таблиці 2, ми бачимо, лінійна нормалізація показує найгірший результат, а нормалізація відношення демонструє значення похибки майже в 4 рази менше для тестової вибірки. Множник 400 для нормалізації відношення був найоптимальнішим для навчальної вибірки.

Аналізуючи графіки на рис. 5 та 6 можна побачити, що всі способи нормалізації відтворюють оригінальні дані зі зсувом вправо, а лінійна нормалізація також демонструє певний зсув вниз. Згідно значень похибки, використання представлення даних з нормалізацією відношення показує в два рази менше значення похибки, а використання різних коефіцієнтів лише покращує значення для тренувальних даних, в той час як значення на тестовому наборі майже не змінюються. Це свідчить про те, що нейронна мережа вчиться "як вчора".

Таблиця 2. Результати середньоквадратичних похибок для лінійної нормалізації та нормалізації відношення.

Table 2. Mean squared error results for linear and ration normalization.

Тип нормалізації	Тренувальні дані	Тестові дані
Лінійна нормалізація	2,3 млн	7,35 млн
Нормалізація відношення з множником 1	1,1 млн	2,09 млн
Нормалізація відношення з множником 36 (стандартне відхилення=1)	0,3 млн	2,88 млн
Нормалізація відношення з множником 400	0,03 млн	2,81 млн

Оскільки нормалізація відношення з множником 400 показала найкращий результат, було вирішено її й залишити. Минулі дослідження використовували тільки максимальне значення ціни протягом часового відтинку. Прийнято рішення створити



Рис. 5. Результати прогнозування нейронних мереж з нормалізаціями відношення: а) весь діапазон даних; б) тренувальна вибірка; в) тестова вибірка.

Fig. 5. Prediction results of neural networks with ratio normalizations: (a) the entire data range; (b) training set; (c) testing set.

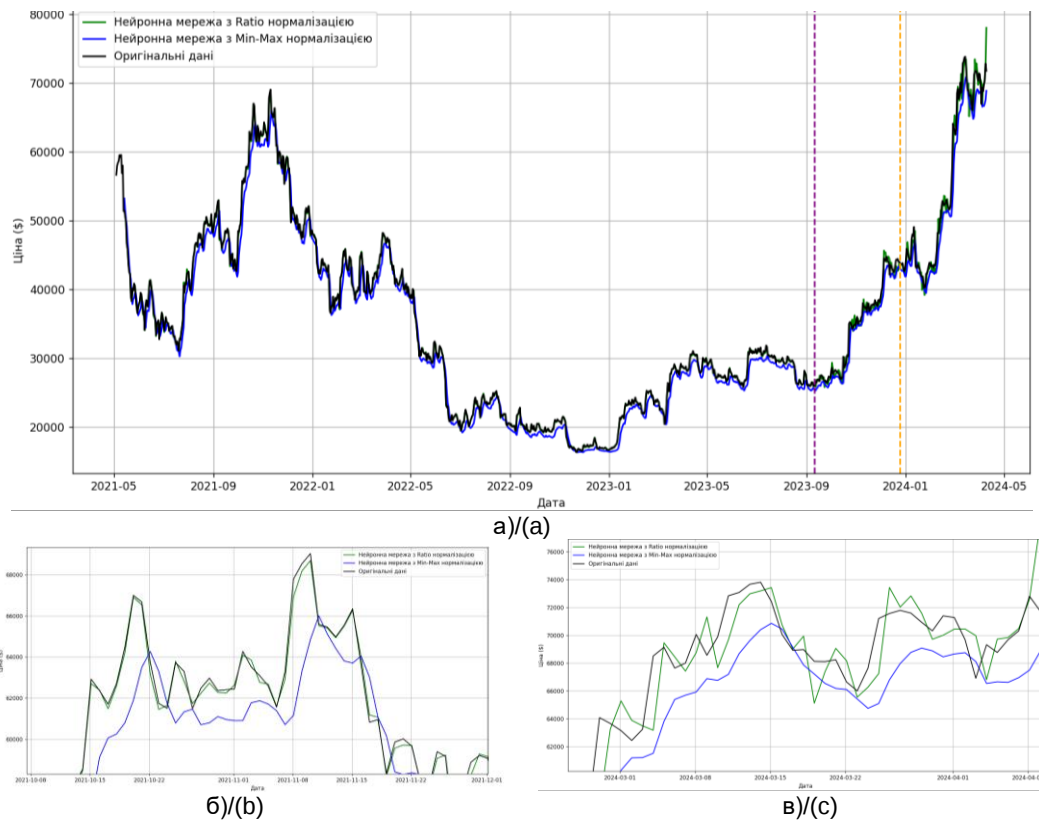


Рис. 6. Результати прогнозування нейронних мереж з лінійною та нормалізацією відношення: а) весь діапазон даних; б) тренувальна вибірка; в) тестова вибірка.
Fig. 6. Prediction results of neural networks with min-max and ratio normalizations: (a) the entire data range; (b) training set; (c) testing set.

нейронну мережу, яка не буде обмежуватись тільки максимальними значеннями ціни, а також використовувати інші характеристики, такі як значення ціни відкриття, закриття та мінімуму протягом часового відліку. Для цього використовуються ті самі дані з біржі Coinbase (за період від 4 травня 2021 року до 9 квітня 2024 року). Нами було створено нейронну мережу, яка обробляє масив значення за 10 днів, який містить 4 колонки значень: максимальної і мінімальної цін, та цін відкриття (початку торгів) і закриття (закінчення торгів), і прогнозує максимальну ціну активу на наступний день. Рис. 7 відображає результати навчання такої нейронної мережі. Дані на графіку поділені на тренувальну, контрольну та тестову вибірки. Дані від початку і до фіолетової прямої відповідають тренувальній вибірці, між фіолетовою та жовтою – контрольні дані, які використовуються для контролю процесу навчання для уникнення перенавчання нейронної мережі і тестові дані.

З графіка видно, що збільшення кількості характеристик не допомогло вирішити проблему з прогнозуванням, оскільки результати прогнозування візуально ніби відстають на 1 часовий відлік, що можна охарактеризувати як тип прогнозування «завтра буде як вчора». Для кращого розуміння проблеми, в таблиці 3 порівнюються значення середньоквадратичних похибок для прогнозованих даних як вони є (з відставанням), та прогнозованих даних, зміщених на один часовий відлік вліво.

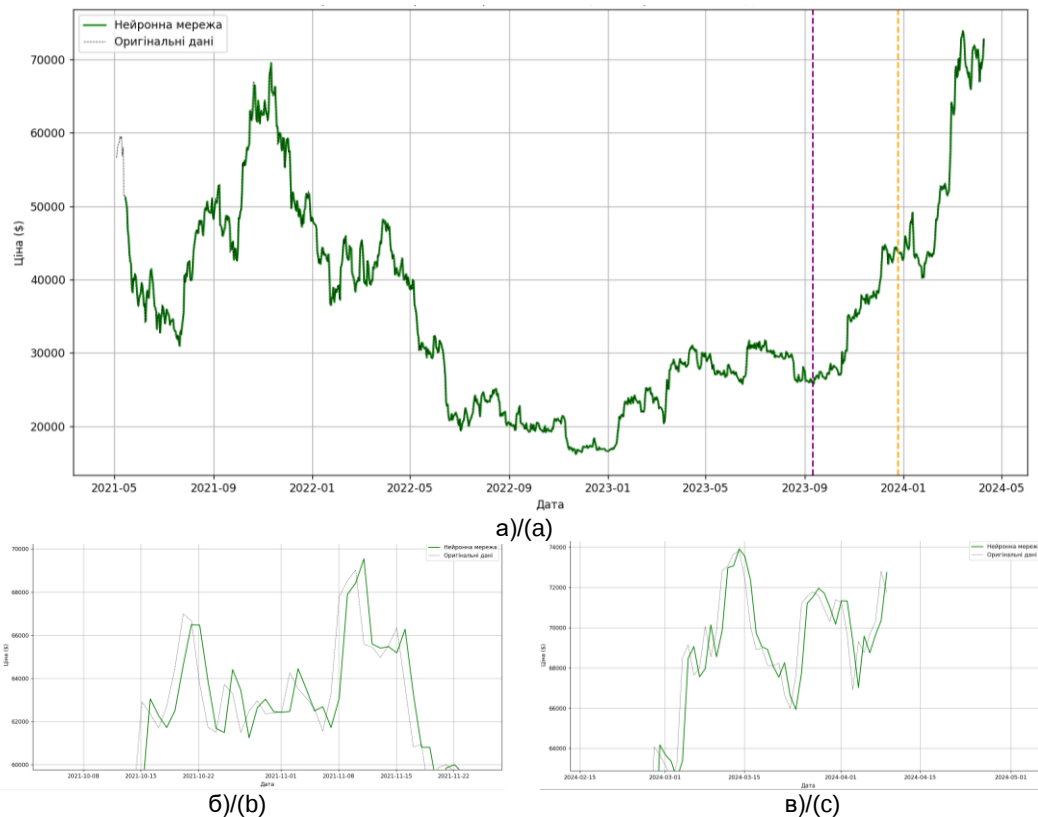


Рис. 7. Результати прогнозування нейронної мережі з нормалізацією відношення, коли на вхід беруться 4 колонки: а) весь діапазон даних; б) тренувальна вибірка; в) тестова вибірка.
Fig. 7. Prediction results of the neural network with ratio normalization when using 4 columns as input: (a) the entire data range; (b) training set; (c) testing set.

Таблиця 3. Порівняння значень середньоквадратичних похибок оригінальних прогнозованих даних та прогнозованих даних після їх зміщення на один часовий відлік вліво.
Table 3. Comparison of mean squared errors of the original forecasted data and the forecasted data after shifting one time step to the left.

Тип даних	Тренувальні дані	Тестові дані
Оригінальні прогнозовані	1,06 млн \$	2,02 млн \$
Прогнозовані дані зміщення на один часовий відлік вліво	0,01 млн \$	0,01 млн \$

ВИСНОВКИ

Таким чином, аналізуючи результати досліджень можна вважати, що використання нейронних мереж для прогнозування ціни віртуальних активів не є ефективним. Така мережа, хоч у певних випадках і показує хороші результати для тренувальних даних, але все ж для тестових даних обирає найкраще рішення наступного елемента як оригінальне попереднє. Нейромережа не справляється із задачею прогнозування ціни, оскільки її передбачення базуються на повторенні

попередніх значень замість справжнього прогнозування. Це свідчить про обмеження моделі в її здатності захоплювати складні взаємозв'язки у даних, що призводить до необхідності пошуку інших підходів або вдосконалення існуючих методів для покращення прогнозування.

ВНЕСОК АВТОРІВ

Концептуалізація, [А.Ц., М.М.]; методологія, [А.Ц., М.М.]; валідація, [М.М.]; формальний аналіз, [А.Ц., М.М.]; дослідження, [А.Ц., М.М.]; написання – підготовка оригінального тексту, [М.М.]; написання – рецензування та редагування, [А.Ц.]; візуалізація, [М.М.] нагляд, [А.Ц.]; адміністративне управління проектом, [А.Ц.].

Усі автори ознайомилися з остаточною версією рукопису та погодилися з його публікацією.

ДЖЕРЕЛА

- [1] Tsemko, A., & Liubun, Z. (2022). USING A NEURAL NETWORK FOR PRICE PREDICTION OF VIRTUAL ASSETS. Electronics and Information Technologies, 20. <https://doi.org/10.30970/eli.20.5>
- [2] Erharter, G. H., & Marcher, T. (2021). On the pointlessness of machine learning based time delayed prediction of TBM operational data. Automation in Construction, 121, 103443. <https://doi.org/10.1016/j.autcon.2020.103443>
- [3] Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2023). Are Transformers Effective for Time Series Forecasting? Proceedings of the AAAI Conference on Artificial Intelligence, 37(9), 11121–11128. <https://doi.org/10.1609/aaai.v37i9.26317>
- [4] Kim, Y.-S., Kim, M. K., Fu, N., Liu, J., Wang, J., & Srebric, J. (2024). Investigating the Impact of Data Normalization Methods on Predicting Electricity Consumption in a Building Using different Artificial Neural Network Models. Sustainable Cities and Society, 105570. <https://doi.org/10.1016/j.scs.2024.105570>
- [5] Coinbase. (n.d.). Pecypc: Coinbase API documentation. Retrieved from <https://docs.cdp.coinbase.com>
- [6] TensorFlow. (2023). Pecypc: TensorFlow guide. Retrieved from <https://www.tensorflow.org/guide>
- [7] TensorFlow. (2024). Pecypc: Time series tutorial. Retrieved from https://www.tensorflow.org/tutorials/structured_data/time_series
- [8] Brownlee, J. (2022, August 7). Time series prediction with LSTM recurrent neural networks in Python with Keras. Retrieved from <https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/>

PROBLEMS OF USING NEURAL NETWORKS TO PREDICT THE PRICE OF VIRTUAL ASSETS

Andrii Tsemko¹  & Maxym Matskiv² 

Ivan Franko National University of L'viv

¹ *Department of Radiophysics and Computer Technologies,*

¹107 Gen. Tarnavskoho St., 79017 Lviv, Ukraine

² *Department of System Design,*

50 Drahomanova St., 79005 Lviv, Ukraine

ABSTRACT

Background. Predicting the prices of virtual assets is an important task due to their high volatility. Neural networks are widely used for such tasks, but often face the problem of naive predictions, when the next value is too similar to the previous one, which reduces the forecasting efficiency.

Materials and Methods. The study was conducted on the basis of Bitcoin price dynamics data (Coinbase exchange) for the period of May 4, 2021 - April 9, 2024. Two methods of data normalization were considered: linear (Min-Max) and ratio normalization, which ensures data stationarity. A neural network with bidirectional LSTM layers was trained on 10 previous values to predict one subsequent value. The TensorFlow library, in particular the Keras API, was used for training process with customizable parameters: 150 epochs, 32 batch size, Adam optimizer, and MSE loss function.

Results and Discussion. Linear normalization showed the worst results, as the model loses its ability to predict if future values fall outside the training set. On the contrary, ratio normalization showed much better results, allowing the model to take into account the dynamics of changes even beyond the minimum and maximum values of the training data. Using an additional multiplier for the normalized data ($k = 400$) improved the results for the training data, although the improvement was not significant on the test set. Adding additional price characteristics (at the opening, closing, and minimum price during this period) failed to eliminate the problem of forecast bias. The analysis of the results showed that the model is often based on the repetition of previous values, which indicates its limitations in capturing complex relationships in the data.

Conclusion. The study confirms that neural networks have limitations in the task of predicting virtual asset prices. The problem of naïve prediction is a key obstacle that limits the effectiveness of models. The use of ratio normalization with a k -factor improves accuracy, but it is not enough to solve the main problem. New approaches or improvements to existing methods are needed to allow the model to better understand complex relationships in time series and reliably predict future changes.

Keywords: recurrent neural networks, forecasting, virtual assets.