https://doi.org/10.31891/csit-2025-3-13
UDC: 519.86:004.8

Oleh PASTUKH, Yuriy PETROV
Ternopil Ivan Puluj National Technical University

EFFICIENCY ANALYSIS OF FINANCIAL TIME SERIES FORECASTING MODELS UNDER MARKET TURBULENCE CONDITIONS

This paper presents a comparative analysis of financial time series forecasting models' effectiveness under market turbulence conditions. The study focuses on evaluating the adaptability of the statistical ARIMA model and the recurrent LSTM neural network across different prediction horizons during periods of high market volatility. Daily OHLC data from five major technology companies (Google, Apple, Amazon, Meta, Oracle) for the period 2020-2025 were analyzed, with particular emphasis on the turbulent April-June 2025 period. Three model architectures were implemented: ARIMA(2,1,0), LSTM Bidirectional Autoencoder (100 units), and simple LSTM (20 units). Testing was conducted across 5-, 15-, and 30-day forecasting horizons using MAPE, RMSE, and MAE metrics. Additionally, residual analysis through autocorrelation function examination was applied to validate model quality. Results demonstrated that ARIMA excelled in short-term forecasts (5 days) with MAPE ≤ 0.06, but its effectiveness diminished on medium-term horizons due to its inability to adapt to market turbulence. The simple LSTM (20 units) achieved an optimal balance between accuracy and stability, outperforming ARIMA by 30.75% on medium- and long-term forecasts. The complex LSTM Autoencoder proved to be the least effective due to overfitting on market noise. The scientific novelty lies in demonstrating that simpler LSTM architectures outperform complex ones under extreme market turbulence conditions, challenging the conventional assumption that model complexity improves performance. Residual analysis was employed as an additional validation method to support these findings. The practical significance includes optimization of algorithmic trading strategies and risk management systems during market instability periods, particularly valuable for financial institutions and investment funds.

Keywords: time series forecasting, ARIMA, LSTM, market turbulence, financial markets, deep learning, volatility, algorithmic trading.

Олег ПАСТУХ, Юрій ПЕТРОВ
Тернопільський національний технічний університет імені Івана Пулюя

АНАЛІЗ ЕФЕКТИВНОСТІ МОДЕЛЕЙ ПРОГНОЗУВАННЯ ФІНАНСОВИХ РЯДІВ В УМОВАХ РИНКОВОЇ ТУРБУЛЕНТНОСТІ

В роботі представлено порівняльний аналіз ефективності моделей прогнозування фінансових часових рядів в умовах ринкової турбулентності. Дослідження зосереджено на оцінці адаптивності статистичної моделі ARIMA та рекурентної нейронної мережі LSTM на різних горизонтах прогнозування під час періодів високої ринкової турбулентності. Проаналізовано щоденні дані формату OHLC п'яти провідних технологічних компаній (Google, Apple, Amazon, Meta, Oracle) за період 2020-2025 років з особливим акцентом на турбулентному періоді квітня-червня 2025 року. Реалізовано три архітектури моделей: ARIMA(2,1,0), LSTM Bidirectional Autoencoder (100 units) та простий LSTM (20 units). Тестування проводилось на горизонтах прогнозування 5, 15 та 30 днів з використанням метрик MAPE, RMSE та MAE. Додатково застосовано аналіз залишків через дослідження функції автокореляції для валідації якості моделей. Результати продемонстрували, що ARIMA перевершує у короткострокових прогнозах (5 днів) з MAPE ≤ 0.06, проте її ефективність знижується на середньострокових горизонтах через неспроможність адаптуватись до ринкової турбулентності. Простий LSTM (20 одиниць) досяг оптимального балансу між точністю та стабільністю, випереджаючи ARIMA на 30.75% на середньо- та довгострокових прогнозах. Складний LSTM Autoencoder виявився найменш ефективним через перенавчання на ринковому шумі. Наукова новизна полягає у доведенні того, що простіші LSTM архітектури перевершують складні в умовах екстремальної ринкової турбулентності, ставлячи під сумнів припущення про те, що складність моделі покращує продуктивність. Аналіз залишків було використано як додатковий метод валідації для підтвердження цих висновків. Практична значимість включає оптимізацію алгоритмічних торгових стратегій та систем управління ризиками в періоди ринкової нестабільності, що має особливе значення для фінансових інститутів та інвестиційних фондів.

Ключові слова: прогнозування часових рядів, ARIMA, LSTM, ринкова турбулентність, фінансові ринки, глибоке навчання, волатильність, алгоритмічна торгівля.

Received / Стаття надійшла до редакції 26.07.2025
Accepted / Прийнята до друку 11.09.2025

Introduction

Effective business management requires accurate assessment of pricing policies and market conditions. Forecasting market behavior becomes particularly critical under extreme conditions such as financial crises and periods of high turbulence.

Financial time series forecasting represents one of the most challenging domains in quantitative finance, especially during periods of market turbulence. The complexity of financial markets is characterized by non-linearity, high volatility, and unpredictable behavior. This necessitates the development of robust and adaptive forecasting models capable of maintaining accuracy under conditions of extreme market stress. Traditional econometric approaches, while theoretically sound, often struggle to capture the dynamic nature of modern financial markets. This limitation becomes particularly evident during crisis periods when historical patterns may become unreliable indicators of future performance.

Related works

Existing research provides comprehensive analysis of various forecasting models performance: autoregressive models - ARIMA, SARIMA, SARIMAX, recurrent neural networks - RNN, GARCH, LSTM, transformers etc. [1; 2].

ARIMA models have long remained the cornerstone of financial time series forecasting, demonstrating effectiveness in capturing linear dependencies and autocorrelations characteristic of financial series. Research conducted by Dariusz Kobiela and Dawid Krefta shows ARIMA superiority by 3.2 times for 30-day forecasts and 1.8 times for 9-month predictions [3]. However, Ruochen Xiao and Yingying Feng conclude that both models are capable of quality forecasts, but LSTM demonstrates better performance, particularly during pronounced stock price changes, while ARIMA application remains more convenient [4].

Meanwhile, comprehensive research by Siami-Namini and Siami-Namin (2018) demonstrated that LSTM-based algorithms show an average error reduction of 84-87% compared to ARIMA models, indicating the superiority of deep learning approaches for certain types of financial data [5]. However, the authors emphasize that deep learning algorithms remain vulnerable to market shocks due to complex causal relationships that make financial markets extremely volatile with sudden and unpredictable price movements.

Martin A. conducts comparative analysis of ARIMA, GARCH, and LSTM models on turbulent data, with conclusions reporting LSTM superiority due to its ability to learn complex patterns [6]. Particular attention in the context of extreme market conditions is drawn to research by Jia and colleagues (2021), who introduced likelihood-based loss functions for LSTM models in volatility forecasting. Their results showed superior performance compared to traditional distance-based loss functions, especially during the March 2020 market shock, when LSTM models better captured extreme market conditions [7].

Contemporary trends in financial forecasting demonstrate growing interest in hybrid approaches. Specifically, Kashif and Ślepaczuk (2024) presented an innovative LSTM-ARIMA model for algorithmic investment strategies, where LSTM serves as the primary forecasting tool while ARIMA estimates and corrects forecasting errors. Testing on three stock indices (S&P 500, FTSE 100, CAC 40) using a 23-year data period that includes two extreme market periods demonstrated the effectiveness of the hybrid approach under high volatility conditions [8].

Contemporary research demonstrates growing interest in advanced hybrid and ensemble methodologies that leverage the complementary strengths of statistical and machine learning approaches. A 2025 Royal Society study proposed an ensemble forecasting procedure integrating LSTM and ARIMA models, demonstrating a significant 15% improvement in root mean square error (RMSE) compared to individual methods, with particular effectiveness during periods of market complexity [9]. Advanced decomposition-based hybrid architectures continue to emerge. Dong and Zhou (2024) introduced a CEEMDAN-SE and ARIMA-CNN-LSTM model that decomposes financial data into stationary high-frequency components (predicted by ARIMA) and low-frequency components (handled by CNN-LSTM networks), achieving superior performance through optimal task allocation between model types [10]. Similarly, recent cryptocurrency forecasting research (2025) using hybrid ARIMA-LSTM for Bitcoin, Litecoin, and Ethereum demonstrates consistent improvements over individual models, with hybrid approaches showing MAE = $726.21 and MAPE = 1.75% compared to standalone LSTM performance [11].

The DLWR-LSTM model (2024) represents another sophisticated hybrid approach, achieving near 1% MAPE for stock market prediction through layered trend separation that effectively captures short-term market dynamics while maintaining robustness across different volatility conditions [12]. SVMD-LSTM hybrid methods (2025) utilizing Successive Variational Mode Decomposition show remarkable accuracy improvements, with up to 73% reduction in error metrics compared to traditional single models, particularly effective for handling the non-stationary characteristics of financial time series [13].

Purpose

A critical research gap persists in understanding optimal model selection and architecture design specifically for extreme market turbulence conditions. While existing studies demonstrate general superiority of various approaches, the specific challenge of rapid model adaptability during crisis periods and the determination of optimal complexity levels for turbulent market forecasting remains underexplored.

This study addresses the key unresolved challenge of model architecture optimization for market turbulence adaptation. Specifically, we examine whether the philosophical principle of Occam's razor applies to financial forecasting during crisis periods. The research aims to determine whether simpler models can outperform complex architectures when markets exhibit extreme unpredictability.

Therefore, the purpose of this study is to analyze the forecasting efficiency of financial time series from the stock market (Google, Apple, Amazon, Meta, Oracle) under turbulence conditions during the April-June 2025 period based on ARIMA and LSTM models**.**

Data preprocessing

The dataset comprises daily OHLC (open, high, low, close) price data for five major technology companies—Google, Apple, Amazon, Meta, and Oracle - retrieved from the Yahoo Finance platform covering the period from January 2, 2020, through May 12, 2025. The complete dataset contains 1,348 observations, representing individual trading day records for each financial time series.

Models were trained using different data volumes to analyze their performance. The dataset was chronologically divided into training and testing samples. Since significant market volatility occurred in early April 2025, the test sample consistently contained data from March 31, 2025, to May 12, 2025 (30 trading days). Two training sample sizes were used: a long training sample covering January 2, 2020, to March 28, 2025 (1,318 rows), and a short training sample covering August 1, 2024, to March 28, 2025 (165 rows).

For the ARIMA model, only univariate data was used, with price forecasts based solely on each company's individual historical data. Given ARIMA's autoregressive nature, the shorter dataset version was employed for training, as autoregressive models cannot capture long-term patterns effectively. In extreme market conditions, relying on more recent data proves more effective.

The LSTM Bidirectional Autoencoder utilized both dataset versions due to its higher parameter count. Insufficient training data leads to underfitting, while excessive epochs cause overfitting. For the simpler LSTM model with 20 units, a short version of the multivariate ensemble dataset was used.

The ensemble dataset combines data from all five companies. When forecasting Google stock prices, information from all companies (Google, Apple, Amazon, Meta, Oracle) is incorporated. This approach offers advantages because stock market performance depends on numerous factors, including company correlations and market interdependencies. Larger companies typically show higher correlation coefficients with overall market movements, meaning significant price changes in one company can trigger responses in others offering similar products or services.

Model Architecture

For the ARIMA model, the parameters were configured as p = 2, d = 1, q = 0, where p = 2 indicates the use of two autoregressive terms to capture the relationship between an observation and its two preceding values, d = 1 represents first-order differencing to achieve stationarity in the time series, and q = 0 indicates no moving average terms were included, suggesting the model relies primarily on autoregressive components for prediction.

The LSTM Bidirectional Autoencoder employed a Seq2Seq architecture with 100 units in each direction, allowing the model to capture both forward and backward temporal dependencies in the data. The input sequence size of 30 was chosen to correspond with approximately one month of trading data, providing sufficient context for pattern recognition. The simpler LSTM model utilized a standard Seq2Seq architecture with 20 units, designed to provide a lightweight alternative with reduced computational requirements. This configuration maintained the same input sequence size of 30 and employed the tanh activation function to ensure consistency in temporal context processing across models. The LSTM Bidirectional Autoencoder was optimized using Mean Squared Error (MSE) loss. The Adam optimizer with a learning rate of 0.001 was chosen for its adaptive learning rate capabilities and efficient convergence properties. Training was conducted for up to 100 epochs with early stopping implemented to prevent overfitting, monitoring validation loss with a patience parameter to ensure optimal generalization. A batch size of 8 was selected to balance computational efficiency with gradient stability, which was particularly important given the model's architectural complexity.

For the Simple LSTM model, Mean Absolute Percentage Error (MAPE) loss was used. The model maintained the same Adam optimizer configuration with a 0.001 learning rate to ensure consistent optimization dynamics. The larger batch size of 16 was feasible due to the model's reduced parameter count, allowing for more stable gradient estimates while maintaining computational efficiency. Early stopping was similarly implemented to achieve the optimal balance between training performance and generalization capability.

Experiments

Testing was conducted across multiple forecast horizons of 5, 15, and 30 days to assess model performance under different temporal prediction challenges.

Table 1

ARIMA predictions

Company	Prediction in days	MAPE	RMSE	MAE
Google	5	0,024	4,46	3,64
Google	15	0,027	4,83	4,06
Google	30	0,03	5,57	4,71
Apple	5	0,059	15,11	12
Apple	15	0,101	22,96	20,96
Apple	30	0,083	18,94	16,34
Amazon	5	0,05	12,11	8,83
Amazon	15	0,075	15,38	13,16
Amazon	30	0,056	12,52	10,12
Meta	5	0,05	36,41	26,57
Meta	15	0,08	50,11	42,9
Meta	30	0,068	43,55	36,68
Oracle	5	0,034	6,23	4,66
Oracle	15	0,061	9,51	7,97
Oracle	30	0,055	9,07	7,53

Table 2

LSTM Bidirectional Autoencoder predictions

Company	Prediction in days	MAPE	RMSE	MAE
Google	5	0.081	13.29	12,28
Google	15	0.078	12,94	11,81
Google	30	0,054	9,71	8,38
Apple	5	0,039	7,51	5,9
Apple	15	0,075	16,81	14,37
Apple	30	0,081	19,82	15,73
Amazon	5	0,051	10,49	9,64
Amazon	15	0,1	20,71	19,33
Amazon	30	0,033	7,96	6,19
Meta	5	0,07	45,42	41,68
Meta	15	0,07	41,71	37,29
Meta	30	0,062	40,43	34,66
Oracle	5	0,04	6,72	5,46
Oracle	15	0,08	12,14	10,89
Oracle	30	0,1	15,97	13,55

Table 3

LSTM 20 units predictions

Company	Prediction in days	MAPE	RMSE	MAE
Google	5	0,02	5,14	3,03
Google	15	0,047	8,71	7,14
Google	30	0,039	7,53	6,01
Apple	5	0,062	13,97	13,11
Apple	15	0,072	16,65	13,82
Apple	30	0,077	19	15,1
Amazon	5	0,061	12,07	11,55
Amazon	15	0,037	8,39	6,87
Amazon	30	0,033	8,21	6,16
Meta	5	0,09	54,34	41,68
Meta	15	0,059	36,52	31,32
Meta	30	0,058	38,68	32,5
Oracle	5	0,036	7,11	4,8
Oracle	15	0,081	11,87	10,57
Oracle	30	0,1	15,84	13,65

Analysis of Results

Analysis of ARIMA model results reveals a distinct pattern of performance variation across data from nearly all examined companies. For short-term forecasting horizons (5 days), ARIMA demonstrates relatively high accuracy, with the MAPE metric remaining below 0.06, corresponding to an average prediction error of 6% from actual prices.

When extending the forecasting horizon to 15 days, MAPE exhibits significant deterioration, increasing by 12.5% for Google and 79.4% for Oracle compared to short-term predictions. Interestingly, long-term forecasting performance (30 days) surpasses medium-term results across all companies except Google, where the MAPE difference between long-term and medium-term forecasts was only 3%.

This performance pattern can be attributed to both the inherent characteristics of the data and ARIMA's fundamental operating principles. The moving average component, combined with the absence of distinct seasonal patterns in the financial data, causes the model's predictions to converge toward a weighted average of historical observations. The observed degradation in medium-term forecasting accuracy corresponds to the period of most severe price volatility, which substantially impacted performance metrics. The subsequent improvement in long-term forecasting metrics reflects market stabilization and price reversion to pre-turbulence trading ranges.

The Autoencoder was the only model trained on both dataset variants. Table 2 presents the results of the model trained on the smaller dataset, as its predictions outperformed those of the model trained on the larger sample in all cases. For short-term forecasting, the LSTM Autoencoder shows worse results compared to ARIMA for 4 out of 5 companies. On average, the short-term Autoencoder forecast underperforms by 46.66%. As the forecast horizon increases, this performance gap decreases: medium-term forecasts underperform by 42.1%, and long-term forecasts by 22.1%.

The simple LSTM shows consistently better results compared to the Autoencoder; therefore, subsequent comparisons focus on the ARIMA model. On average, ARIMA's short-term forecast is 19.26% more accurate than LSTM, with the difference reaching as high as 80% for Meta company data. These results confirm ARIMA's effectiveness for short-term forecasts. The medium-term LSTM forecast showed better results in 3 out of 5 cases. When comparing the graphs of medium-term forecasts across all models, LSTM can be distinguished by its superior recognition of momentum shifts (Fig. 1, Fig. 2).

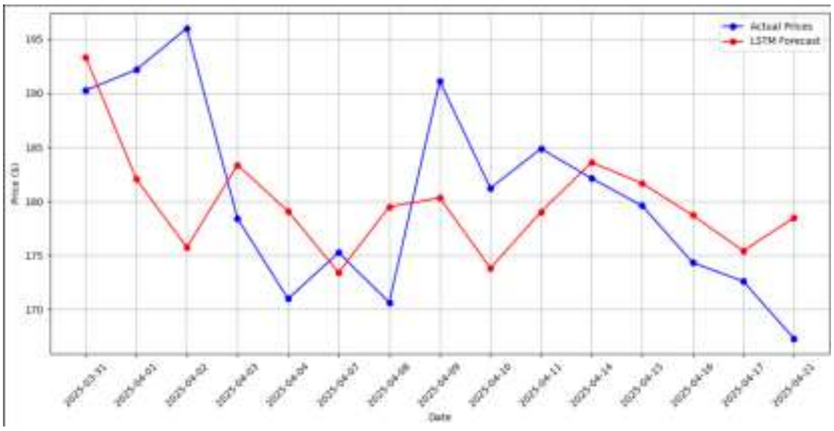

Fig. 1. LSTM 20 units middle horizon (15 days) Amazon stock price prediction

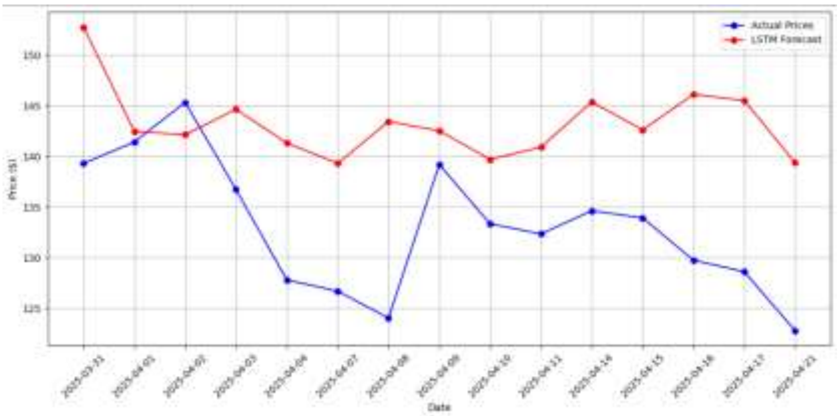

Fig. 2. LSTM 20 units middle horizon (15 days) Oracle stock price prediction

The long-term LSTM forecast also showed better results in 3/5 cases. Unlike ARIMA, LSTM results do not deteriorate sharply depending on market volatility (Fig. 3).

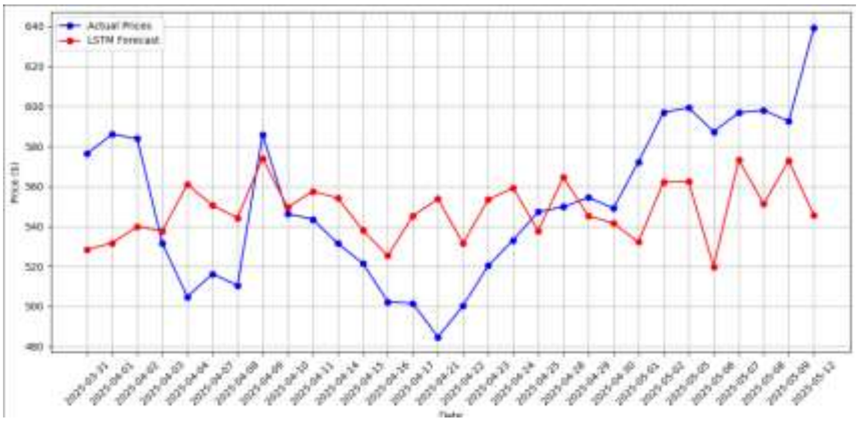

Fig. 3. LSTM 20 units long horizon (30 days) Meta stock price prediction

Residual Analysis

Model quality was assessed through analysis of the autocorrelation function (ACF) of residuals. A well-performing model should produce residuals resembling white noise without systematic patterns, reflected in the absence of significant autocorrelations at all lags.

Short-term forecast (5 days). All investigated models demonstrated statistically insignificant autocorrelations in residuals, indicating effective detection of dependencies in the data. The simple LSTM exhibited the closest approximation to white noise in residuals, suggesting the model's superior ability to capture latent dependencies and temporal patterns in the time series (Fig. 4)

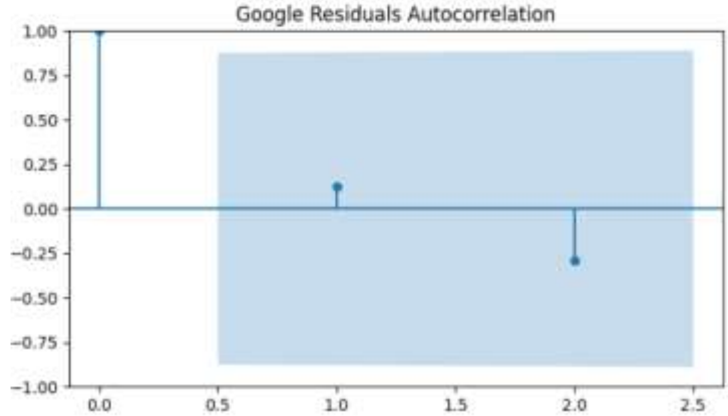

Fig. 4. Autocorrelation function residuals of the short-term forecast of the LSTM model 20 units

Medium-term forecasts (15 days). The LSTM Autoencoder (Fig. 5) and ARIMA began to demonstrate weak but noticeable autocorrelation spikes, indicating incomplete capture of complex nonlinear dependencies. The simple LSTM better preserved white noise characteristics in residuals, confirming its robustness across increasing forecast horizons.

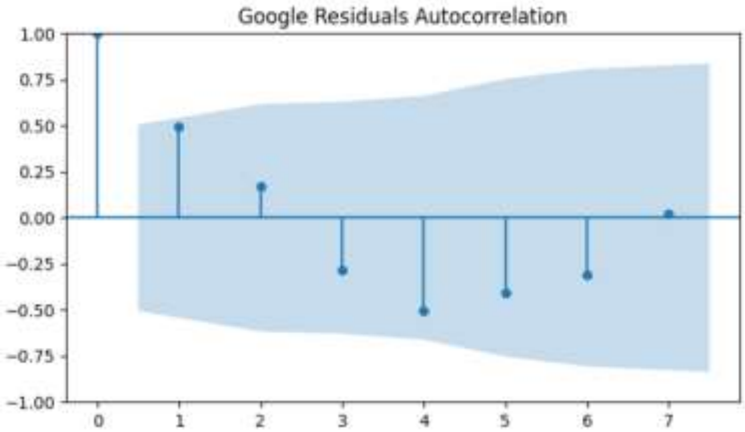

Fig. 5. Autocorrelation function residuals of the middle-term forecast of the LSTM Autoencoder

Long-term forecasts (30 days). The results of long-term forecast analysis deteriorated across all models. ARIMA (Fig. 6) and the LSTM Autoencoder left the most systematic structure in residuals. The simple LSTM demonstrated the best residual characteristics, closest to white noise, indicating an optimal balance between model complexity and generalization capability.

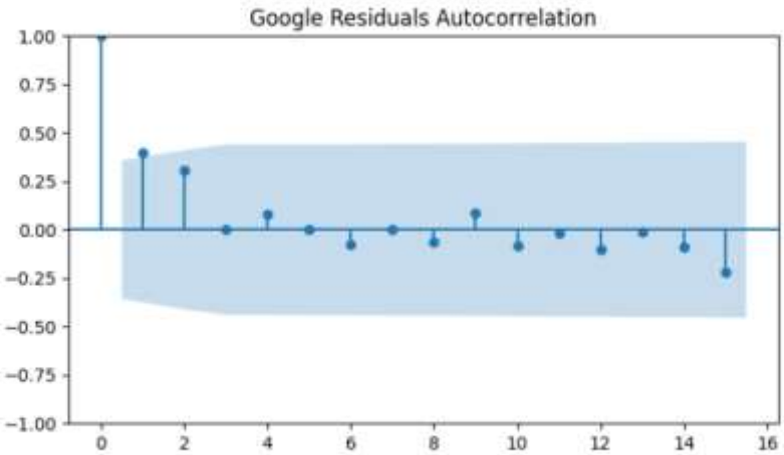

Fig. 6. Autocorrelation function residuals of the long-term forecast of the LSTM model 20 units

Conclusions

Financial time series of the stock market exhibit pronounced stochastic characteristics. Increasing the training sample size (length of temporal context) causes an increase in forecast error, as the model begins to identify dependencies in random noise. Complex models, such as the LSTM Autoencoder, require significantly more time and computational resources for training due to their large number of parameters. Despite this complexity, the Autoencoder demonstrates less accurate results compared to simpler ARIMA and LSTM 20-unit models.

This behavior stems from the large number of parameters relative to the limited amount of data. The information content in the dataset becomes highly dispersed among model weights, introducing considerable noise in the final results. Consequently, the model captures only the general averaged trend without accounting for short-term fluctuations. Residual analysis results confirm that increasing the number of parameters does not improve, but rather worsens, the model's ability to detect dependencies in noisy data.

ARIMA demonstrated strong results for short-term forecasts despite its algorithmic simplicity and relatively low hardware requirements for training. This makes autoregressive models a suitable choice for forecasting financial series under lower turbulence conditions. However, ARIMA cannot process multivariate data and struggles to recognize turning points in noisy data without clear seasonal patterns. ARIMA captures linear dependencies, while financial markets often exhibit nonlinear effects. Residual analysis indicates that ARIMA captures fewer patterns compared to recurrent networks.

The LSTM 20-unit model represents the optimal balance between capturing relevant patterns and avoiding overfitting on market noise. Residual analysis confirms the model's ability to understand data structure across all forecast horizons. The reduced parameter count requires fewer computational resources for training, while the multivariate dataset enables effective forecasting of turning points with minimal delay and faster adaptation to sharp market changes under turbulent conditions.

On average, LSTM 20 units shows 26.2% more accurate results compared to the LSTM Autoencoder. A particularly significant difference occurred with Google company data, where the simple LSTM outperformed the Autoencoder by 58.7% on average across all horizons. For medium- and long-term forecasts, LSTM 20 units outperforms ARIMA by an average of 30.75%.

References

1. Box G.E.P., Jenkins G.M. Time Series Analysis: Forecasting and Control. San Francisco: Holden-Day, 1970. 553 p.
2. Hochreiter S., Schmidhuber J. Long Short-Term Memory. Neural Computation. 1997. Vol. 9, No. 8. P. 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735.
3. Kobiela D., Krefta D. ARIMA vs LSTM on NASDAQ stock exchange data. Procedia Computer Science. 2022. Vol. 207. P. 3836-3845. https://doi.org/10.1016/j.procs.2022.09.443.
4. Xiao R., Feng Y. Stock price prediction using ARIMA and LSTM models. International Journal of Financial Studies. 2021. Vol. 9, No. 3. P. 42. https://doi.org/10.3390/ijfs9030042.
5. Siami-Namini S., Siami Namin A. Forecasting Economics and Financial Time Series: ARIMA vs. LSTM. arXiv preprint. 2018. arXiv:1803.06386. https://doi.org/10.48550/arXiv.1803.06386.
6. Martins A., Forecasting Volatility: Comparative Analysis of ARIMA, GARCH, and Deep Learning Models for Predicting Stock Market Volatility / M. Singh et al. Research Gate. 2023
7. Jia M., Li X., Wang L. Forecasting Volatility of Stock Index: Deep Learning Model with Likelihood-Based Loss Function. Complexity. 2021. Vol. 2021. P. 5511802. https://doi.org/10.1155/2021/5511802.
8. Kashif K., Ślepaczuk R. LSTM-ARIMA as a Hybrid Approach in Algorithmic Investment Strategies. arXiv preprint. 2024. arXiv:2406.18206. https://doi.org/10.48550/arXiv.2406.18206.
9. Royal Society Open Science. An ensemble approach integrating LSTM and ARIMA models for enhanced financial market predictions. 2025. https://doi.org/10.1098/rsos.240699.
10. Dong, Z.; Zhou, Y. A Novel Hybrid Model for Financial Forecasting Based on CEEMDAN-SE and ARIMA-CNN-LSTM. Mathematics. 2024. Vol. 12, No. 16. P. 2434. https://doi.org/10.3390/math12162434.
11. HighTech and Innovation Journal. Closing Price Prediction of Cryptocurrencies BTC, LTC, and ETH Using a Hybrid ARIMA-LSTM Algorithm. 2025. Vol. 6, No. 2. Available online: https://hightechjournal.org/index.php/HIJ/article/view/1170
12. ScienceDirect. Time series forecasting of stock market indices based on DLWR-LSTM model. 2024. Available online: https://www.sciencedirect.com/science/article/abs/pii/S1544612324008511
13. Wiley Online Library. Time-Series Forecasting Using SVMD-LSTM: A Hybrid Approach for Stock Market Prediction. Journal of Probability and Statistics. 2025

Oleh Pastukh **Олег Пастух**	Dr.Sc, Professor, Professor of Department Software Engineering, Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine, e-mail: oleg.pastuh@gmail.com https://orcid.org/0000-0002-0080-7053	доктор наук, професор, професор кафедри програмної інженерії Тернопільського національного технічного університету імені Івана Пулюя, м. Тернопіль, Україна
Yuriy Petrov **Юрій Петров**	Bachelor student of Department Software Engineering, Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine, e-mail: petrov.jurij.s@gmail.com https://orcid.org/0009-0003-1045-0486	студент-бакалавр кафедри програмної інженерії Тернопільського національного технічного університету імені Івана Пулюя, м. Тернопіль, Україна