UDC 004.8:519.23:336.76
DOI https://doi.org/10.24144/2616-7700.2025.47(2).168-176

N. E. Kondruk[1], S. V. Hetsko[2]

[1] Uzhgorod National University,
Associate Professor of Department of Cybernetics and Applied Mathematics,
PhD in Technical Sciences
natalia.kondruk@uzhnu.edu.ua
ORCID: https://orcid.org/0000-0002-9277-5131

[2] Uzhgorod National University,
Postgraduate student of Department of Cybernetics and Applied Mathematics
serhii.hetsko@uzhnu.edu.ua
ORCID: https://orcid.org/0000-0001-9163-5279

DEVELOPMENT OF MULTI-TIMEFRAME MACHINE LEARNING-BASED DECISION SUPPORT SYSTEMS FOR ALGORITHMIC FOREX TRADING

The effectiveness of decision support systems (DSS) in the foreign exchange market (FOREX) critically depends on the selected timeframe and machine learning objective function (ML-objective). The need to determine the optimal combination of these factors determines the relevance of this study. The aim of the work is to develop and conduct a comparative analysis of the performance of three DSS architectures (Systems A, B and C) to establish the optimal mathematical formulation of the market forecasting problem.

Three architectures were implemented: System A is based on regression forecasting; System B uses classification based on features of a single timeframe (15M); System C uses classification using multi-timeframe features. The models were trained on historical data, and their effectiveness was verified by backtesting on an independent (OOS) data set.

The impact of multi-timeframe features and ML-objective on DSS performance for the XAU/USD asset was studied. According to the results of the comparative analysis, System C demonstrated exceptional results, surpassing other architectures: the total return was 3283.69%, and the maximum drawdown was only 2.07%. Prospects for further research include: developing an integrated risk management methodology based on ML models; implementing early position closing mechanisms (in particular, Trailing SL); and improving the DSS to compensate for model degradation.

Keywords: algorithmic trading, FOREX, Machine Learning, Forecasting, Classification, LSTM.

1. Introduction. Global Financial Markets serve as the primary engine for capital allocation. The challenges associated with these markets involve high-dimensional time-series, positioning them among the most challenging tasks in modern computational science. The Foreign Exchange (FOREX) market, being the largest and most liquid of all global markets, is fundamental to the stability and operation of the global system.

While traditional methodologies, such as technical analysis (TA) and fundamental analysis (FA), remain industry standards – they often result in subjective and heuristic frameworks rather than deterministic decision-making systems [1]. ML-models that are built on basis of TA and FA, allowing them to retain their explanatory power, while mitigating inherent human limitations. Unlike humans, autonomous ML systems do not require rest, are devoid of subjectivity and emotional bias, and can be trained on substantial amount of data. Furthermore, they can be rigorously backtested and iteratively optimized, which is significant advantage over discretionary systems. As a result, algo-trading based on ML models has gained widespread adoption.

The object of the study is non-stationary financial time-series data of foreign Exchange Market (FOREX), specifically represented by OHLC. The subject of the study is the influence of ML objective function on the practical efficiency and utility of automated trading decision support systems.

A significant body of research has explored quantitative models and methods in financial markets. Nevertheless, these studies often focus predominantly on forecasting and neglect the classification approach. Even in instances where classification models are proposed, their comparison with regression-based forecasting models is often overlooked. When developing these models, researchers typically prioritize minimizing continuous error metrics such as Root Mean Squared Error (RMSE). Moreover, these works often fail to pay sufficient attention to multi-timeframe analysis. Even though, multiple studies demonstrate that models with lower RMSE do not always yield commensurately higher trading returns or better risk management [2–4]. Moreover this objective is fundamentally misaligned with main practical goal of trading. Our work is focused on investigating the underlying correlation between ML objective function and model performance. Additionally, we explore the impact of multi-timeframe features on efficiency. By analyzing the effectiveness of these three distinct systems, we aim to provide a more nuanced understanding of their predictive capabilities. This paper will delve into the comparison of three distinct decision support systems: System A, which utilizes a forecasting model as signal generator; System B, which employs classification model with single time-frame features, System C, which employs classification model with multi-timeframe features.

The purpose of this work is to develop and comparatively analyze the performance of three distinct algorithmic trading decision-support systems (A, B, and C) to determine the optimal mathematical framing of the market prediction problem.

To achieve the purpose of the work, the following problems shall be solved:

– create, and preprocess the datasets;
– design, implement and train machine learning models;
– develop and integrate decision-support system prototypes;
– conduct comprehensive backtesting, performance evaluation, and comparative analysis.

2. Problem Statement. Let the multi-variate time series of market data be represented as $X = \{x_1, x_2, \ldots, x_T\}$, where $x_i \in \mathbb{R}^n$ (n features).

A single trade can be modeled as a decision function f that processes the m most recent observations and outputs a discrete action: $a_{t_i} = f\left(x_{t_{i-m}}, x_{t_{i-m+1}}, \ldots, x_{t_{i-1}}\right)$.

Where the set of possible discrete actions is $A = \{-1, 0, 1\}$, $a_{t_i} \in A$. Specifically: -1 — corresponds to a signal to open short position, 0 — signals a null action (do nothing), and 1 — signals the opening of a long position.

The outcome of a single trade depends on action a_{t_i} (entry at t_i, exit at $t_i + k_i$) and the subsequent market data: $x_{t_i}, x_{t_{i+1}}, \ldots, x_{t_{i+k}}$. The outcome function g maps action and future data to the resulting trade value (profit/loss): $r_{t_i} = g\left(a_{t_i}, x_{t_i}, x_{t_i+1}, \ldots, x_{t_i+k_i}\right)$, where r_{t_i} is the return for the trade initiated at time t_i, and $g : A \times (\mathbb{R}^n)^* \to \mathbb{R}$.

Total performance metric is defined as function h, aggregating the results of all trades taken over the observation window: $P = h\left(r_{t_1}, r_{t_2}, \ldots, r_{t_N}\right)$.

Where P represents the overall profitability or a risk-adjusted return metric. Therefore the goal is to maximize the performance metric P by optimizing the decision function f:

$$\underset{f \in F}{max}\, P = h\left(\left(g\left(f\left(x_{t_i-m}, x_{t_i-m+1}, \ldots, x_{t_i-1}\right), x_{t_i}, x_{t_i+1}, \ldots, x_{t_i+k_i}\right)\right)_{i=1}^{N}\right),$$

where $F \subseteq \{f : (\mathbb{R}^n)^m \to A\}$.

Consequently the research objective is to construct the optimal model and decision making system that maximizes the defined performance metric.

3. Review of the literature. Studies [5, 6] have investigated the application of GRU, RNN, LSTM to financial time-series forecasting. However, this body of work focused strictly on improving predictive accuracy, and did not prioritize the real-world trading metrics.

Subsequent research has compared ARIMA and LSTM [7], examined different architectures, including Random Forest and LSTM [8], and evaluated novel structures like single versus dual LSTM [9]. Critically, all of these studies [7–9] primarily assess performance solely in terms of minimizing predictive error, a methodology which is demonstrably insufficient.

As evidenced by numerous investigations [2–4], the RMSE is and inherently deficient metric for evaluating trading models. Because it fails to accurately reflect practical trading objectives [2, 4], overlooks profitability and directionality [2, 4], and is highly susceptible to market noise [3].

While minority of studies [10, 11] have explored use of classification framework instead of forecasting. It lacks direct comparative analysis between the regression and classification methodologies.

This necessitates exploring correlation between ML objective function and model performance in algorithmic trading. Our study will compare: System A would use forecasting model as signal generator, System B would use classification model with single time-frame features, System C would use classification model with multi-timeframe forecasting features.

4. Materials and Methods. LSTM model. Long Short-Term Memory (LSTM) networks [9] are a type of neural network that is capable of learning from long sequences of data. LSTMs have memory that helps them remember important information from past data. This allows them to build connections on how past events impact future.

LSTM has forget gates, input gates, and output gates. These gates control how information flows inside the LSTM's memory. LSTM can decide which information to keep, which to forget, and how to combine it with new data to make better predictions.

Normalization is used to bring data to a common scale. In this study, the min-max technique (1) was employed:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}. \tag{1}$$

All development were done using Python [12]. Python is high-level programming language that is industry standard for Machine Learning. It became industry standard because it's easy to read with simple syntax and rich ecosystem of libraries that utilizing C++ for performance. Pandas library was used for data storing, manipulation, cleaning. For ML-models were used TensorFlow / Keras [13]. Matplotpib was used for visualisation.

5. Experiments.

1. Datasets.

The primary financial asset investigated was XAU/USD.

We trained regression LSTM model for System A on Dataset A, which consists of 124 176 rows. It was gathered from the finance.yahoo.com website.

Dataset A was collected on 15 minutes time-frame. And includes data for time period from 01 January 2020 to 31 December 2024. Dataset has 8 columns. First column labeled as Date Time serving as an index. The following four columns labeled as Open, High, Low, Close were used as input features.

We trained classification LSTM models (System B and System C) on dataset of trades. Let's call it Dataset B. Dataset B were data-mined from Dataset A.

Dataset B consists of 124 176 rows. And following columns: Entry time — time when trade was entered, also serving as an index; entry price; open; high; low; close; exit time — time when trade was closed; direction — direction of trade(either 1(Long) or -1 (Short)); exit price — price when trade was closed; win — binary outcome; profit — return magnitude. All columns except typed as "datetime", win and profit — are features. Win — is target variable.

Data were normalized to a range of [0, 1] using the Min-Max normalization technique as shown in (1).

All datasets were divided into In Sample (IS) — used for training and tuning of model and Out of Sample (OOS) — used for evaluation of model (backtesting). IS includes data for time period from 01 January 2020 to 31 December 2023. OOS includes data for time period from 01 January 2024 to 31 December 2024. IS were divided into training set (first 70% rows) and test set (last 30% rows).

2. Scheme of experiment.

Gathering Dataset A. Dataset A was acquired by utilizing "yfinance" Python library.

Creating Dataset B. Dataset B was generated through a data-mining process applied to the raw price data of Dataset A. For each data point, a hypothetical trade was initiated with predefined SL equal 2 × ATR and TP equal 6 × ATR. Initial direction was selected randomly. The outcome of each trade—recorded as either a win or loss—was then documented, and the process advanced to the subsequent data point.

System A. Dataset A was utilized to train a forecasting model based on LSTM architecture as referenced in [1]. The model implementation was conducted using TensorFlow library.

Decision Support System A was subsequently developed to utilize Model A as signal source. On the first tick of each bar Model A generates a prediction for high low and close of this bar based of prior bars. Based on this forecasted price data system executes one of three discrete actions: "Enter Long", "Enter Short" or "Does nothing".

System B. As a preparatory step, Dataset B was enriched with and array of trend, momentum, and volatility indicators derived from the 15-minute OHLC data.

Subsequently, an LSTM classification model (Model B) was trainedon this augmented dataset. Early Stopping was employed to mitigate risk of overfitting.

Decision Support System B uses model B as its signal source. At the beginning of each bar all features are passed to the Model B. To determine the directional action, Model B performs two separate classifications (one for Long and one for Short), yielding a probability of success (P) for each direction.

The activation rule is governed by a confidence threshold (Θ). If the probability of a win exceeds the threshold for only one direction, a trade is initiated in that direction. If the probability of a win exceeds the threshold for both directions, the trade is opened in the direction with the higher probability. If the probability of a win falls short of the threshold for both directions, the system executes a null action (does nothing).

The theoretical minimum threshold, derived via Expected Value optimization given our risk-to-reward ratio of 1:3, is 0.25. However, to ensure robust performance accounting for transaction costs, model uncertainty, and risk-adjusted returns, a conservative threshold of 0.5 was selected. This choice optimally balances the trade-off between expected return and variance: a lower threshold would significantly increase position variance and exposure to unnecessary risk, whereas a higher threshold would unacceptably diminish the opportunity set. The Stop Loss (SL) and Take Profit (TP) levels are set identically to the parameters utilized during the generation of Dataset B.

System C. System C represents an architectural extension of System B, developed with an identical modeling approach but with a significantly expanded feature set. System C incorporates features calculated based on the current 15-minute timeframe in conjunction with technical indicators derived from higher timeframes (1-Hour and 4-Hour).

Implementing Backtesting framework. The specific computational requirements of this comparative analysis necessitated the development of a proprietary backtesting framework. Existing backtesting solutions were insufficient due to performance limitations and an inability to incorporate the non-standard architectures of Systems B and C.

Upon completion of all prerequisite, backtesting was executed on OOS data. All system simulations were initialized with initial balance of $100,000 and fixed Risk-Per-Trade of 0.5%.

6. RESULTS. Table 1 presents the results obtained for the Systems A,B and C on OOS.

Table 1.

DSS backtesting results

	System A	System B	System C
Final Balance	$109,083.41	$207,275.17	$3,383,686.07
Total Return	9.08%	107.28%	3283.69%
Win Rate	45.98%	38.66%	67.08%
Profit Factor	1.18	1.03	4.87
Max Drawdown	5.76%	66.91%	2.07%

In Figures 1-3 compared results of each of Systems A, B and C.

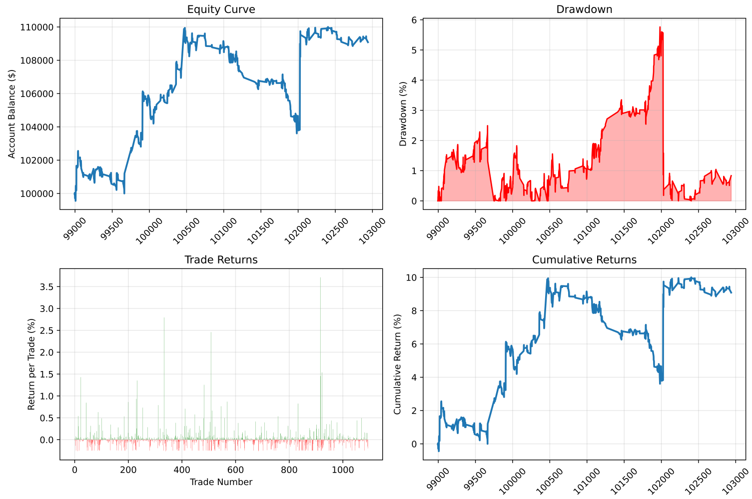


Figure 1. Equity curve, drawdown, trade returns and cumulative returns of System A.

The Equity Curve for System A demonstrates the capacity for sustained capital appreciation, a finding strongly supported by the Cumulative Returns chart. However the system's inherent risk profile is linked to significant, periodic drawdowns as evidenced by the Drawdown chart Furthermore.

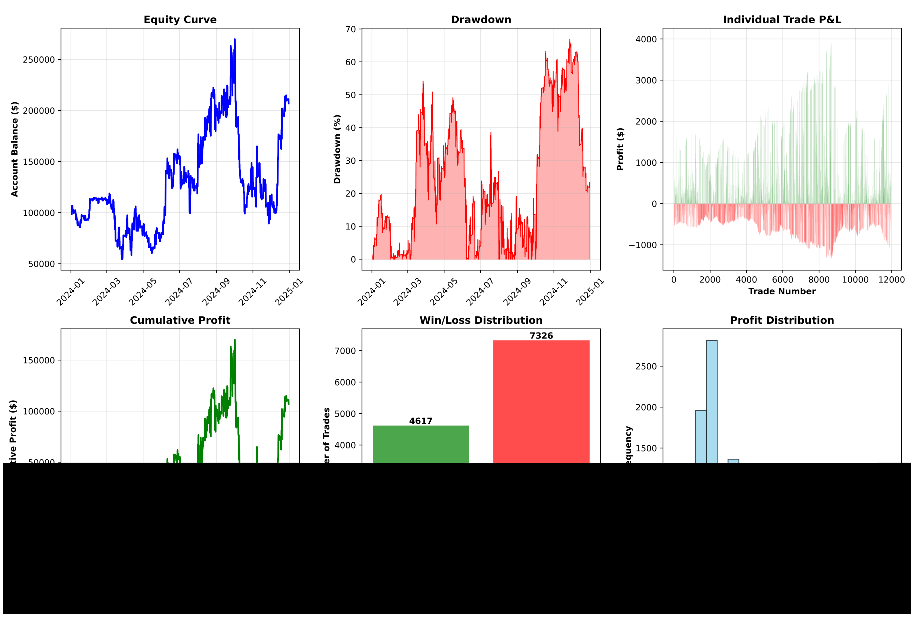
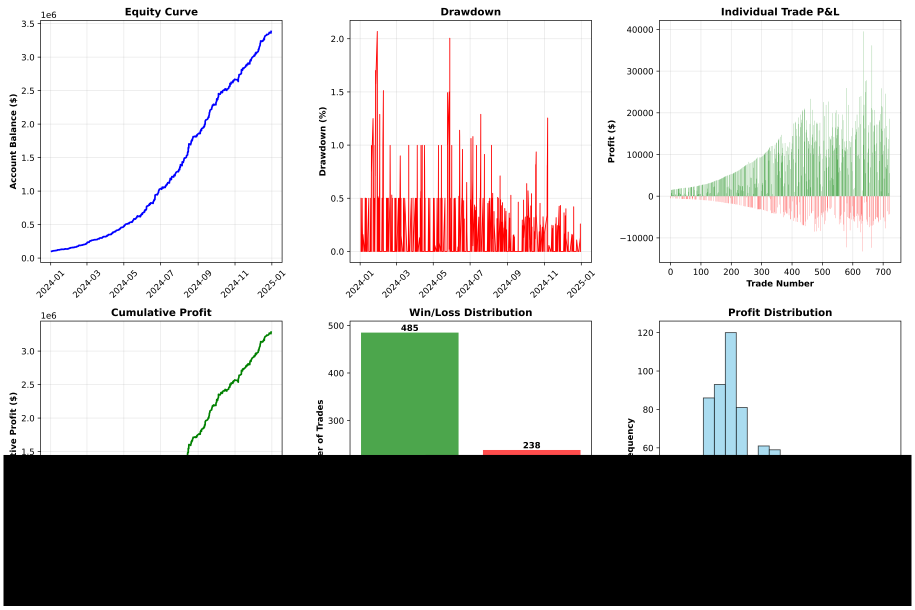


Figure 2. Equity curve, drawdown, trade returns and cumulative returns of System A.

The Equity Curve and Cumulative Profit metrics for System B exhibit characteristics analogous to those observed in System A. However, System B demonstrates a superior capacity for capital growth, achieving higher returns. This enhanced profitability is, accompanied by a higher degree of aggressiveness and increased frequency of periodic drawdowns, followed by rapid recovery cycles.



[redacted]

Figure 3. Equity curve, drawdown, trade returns and cumulative returns of System C.

The Equity Curve and Cumulative Profit plots for System C unequivocally demonstrate steady, exponential capital growth. The system achieves profitability levels that exceed, the aggression demonstrated by System B. Crucially, however, the Drawdown chart confirms that the maximal drawdown was constrained to approximately 2%, with all drawdowns characterized by rapid recovery cycles. Overall, System C successfully couples high-magnitude profit generation (similar to System B) with superior risk containment (manifested in faster recovery and a significantly smaller maximal drawdown), validating its optimization as the most efficient architecture.

7. DISCUSSIONS. Although System A, operating under conventional regression paradigm, achieved a competitive MAE equal to 0.0337. The comparative analysis demonstrated the superior efficacy of the classification-based approaches. Specifically, Systems B and C exhibited metrics that are demonstrably more aligned with real-world objective of trading (Table 1). While System A is capable of generating profit, its equity curve (Figure 1) is overly volatile, yielding relatively modest returns. System B (Figure 2), which utilizes the classification ML model, generates substantially higher returns, though this is accompanied by elevated drawdowns. Ultimately, System C, which integrates multi-timeframe features into the classification, is confirmed to deliver highly stable results characterized by minimal drawdowns (Figure 3).

8. CONCLUSIONS. The problem of development and comparative analysis of the performance of three distinct algorithmic trading DSS (A, B, C) to determine the optimal mathematical framing of the market prediction has been solved.

The scientific novelty of the obtained results is summarized as follows:

1. Systems A, B and C were applied to the algo-trading of the XAU/USD currency pair at 15M timeframe;
2. the hyperparameters of the models and the architecture of neural networks were adjusted in accordance with the tasks under consideration;
3. to identify the most effective ML objective function, a critical analysis of the results was conducted, encompassing both visualisations and financial metrics;
4. according to the results of the study, the best system was System C with Total Return — 3283.69%, Win Rate — 67.08% and Max Drawdown — 2.07%.

The practical significance of our findings lies in the development of three DSS: System A, System B, System C. Experiments have demonstrated that framing trading as classification problem instead of forecasting problem leads to improving performance metrics of resulting DSS. This software can empower traders with the ability to make better decisions and therefore maximizing profitability. Furthermore, the models and the underlying methodology presented in this work can serve as valuable resources for other researchers seeking to advance the field of financial forecasting.

Future research is to develop a method for Risk and Money Management with ML-models. Also future researches could focus on implementing DSS that would account for early stopping, and exiting position before reaching SL/TP, or implementing Trailing SL. It is also planned to enhance current decision support system to account for model decay.

References

1. Kondruk, N. E., & Hetsko, S. V. (2024). Forecasting currency rates using machine learning models. *Scientific Bulletin of Uzhhorod University. Series of Mathematics and Informatics,* 45(2), 205–215. `https://doi.org/10.24144/2616-7700.2024.45(2).205-215`
2. Shah, J., Vaidya, D., & Shah, M. (2022). A comprehensive review on multiple hybrid deep learning approaches for stock prediction. *Intell. Syst. Appl.,* 16, 200111. `https://doi.org/10.1016/j.iswa.2022.200111`
3. Sattar, A., Sarwar, A., Gillani, S., Bukhari, M., Rho, S., & Faseeh, M. (2025). A Novel RMS-Driven Deep Reinforcement Learning for Optimized Portfolio Management in Stock Trading. *IEEE Access,* 13, 42813–42835. `https://doi.org/10.1109/ACCESS.2025.3546099`
4. Liu, F., Guo, S., Xing, Q., Sha, X., Chen, Y., Jin, Y., & Zheng, Q. (2024). Application of an ANN and LSTM-Based Ensemble Model for Stock Market Prediction. In *2024 IEEE 7th International Conference on Information Systems and Computer Aided Education (ICISCAE),* 390–395. `https://doi.org/10.1109/ICISCAE62304.2024.10761432`
5. Sako, K., Mpinda, B. N., & Rodrigues, P. C. (2022). Neural networks for financial time series forecasting. *Entropy,* 24(5), 657. `https://doi.org/10.3390/e24050657`
6. Alfredo, C. S., & Adytia, D. A. (2022). Time series forecasting of significant wave height using GRU, CNN-GRU, and LSTM. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi),* 6(5), 776–781. `https://doi.org/10.29207/resti.v6i5.4160`
7. Taslim, D. G., & Murwantara, I. M. (2022). A Comparative Study of ARIMA and LSTM in Forecasting Time Series Data. In *2022 9th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE).* Semarang: Indonesia, 231–235. `https://doi.org/10.1109/ICITACEE55701.2022.9924148`
8. Ghosh, P., Neufeld, A., & Sahoo, J. K. (2021). Forecasting directional movements of stock prices for intraday trading using LSTM and random forests. *Finance Research Letters,* 102280. `https://doi.org/10.1016/j.frl.2021.102280`
9. Gajamannage, K., Park, Y., & Jayathilake, D. (2023). Real-time forecasting of time series

in financial markets using sequentially trained dual-LSTMs. *Expert Syst. Appl.,* 223, 119879. https://doi.org/10.1016/j.eswa.2023.119879
10. Dezhkam, A., Manzuri, M., Aghapour, A., Karimi, A., Rabiee, A., & Shalmani, S. (2022). A Bayesian-based classification framework for financial time series trend prediction. *The Journal of Supercomputing,* 79, 4622–4659. https://doi.org/10.1007/s11227-022-04834-4
11. Chalkidis, N., & Savani, R. (2021). Trading via selective classification. *Proceedings of the Second ACM International Conference on AI in Finance.* https://doi.org/10.1145/3490354.3494379
12. Welcome to python.org. Retrieved from https://www.python.org
13. TensorFlow. Retrieved from https://www.tensorflow.org

Кондрук Н. Е., Гецко С. В. Розробка систем підтримки прийняття рішень на основі машинного навчання з мульти-таймфреймними даними для алгоритмічної торгівлі на ринку Форекс.

Ефективність систем підтримки прийняття рішень (СППР) на ринку іноземних валют (FOREX) критично залежить від обраного таймфрейму та цільової функції машинного навчання (ML-ціль). Необхідність визначення оптимального поєднання цих факторів обумовлює актуальність даного дослідження. Метою роботи є розробка та здійснення порівняльного аналізу продуктивності трьох архітектур СППР (Системи A, B та C) для встановлення оптимального математичного формулювання задачі прогнозування ринку.

Було імплементовано три архітектури: Система A базується на регресійному прогнозуванні; Система B використовує класифікацію на ознаках одного таймфрейму (15M); Система C застосовує класифікацію з використанням мульти-таймфреймних ознак. Навчання моделей проведено на історичних даних, а їхня ефективність верифікована шляхом бектесту проведеного на незалежному (OOS) наборі даних.

Проведено дослідження впливу мульти-таймфреймних ознак та ML-цілі на продуктивність СППР для активу XAU/USD. За результатами порівняльного аналізу, Система C продемонструвала виняткові результати, перевершивши інші архітектури: загальна прибутковість склала 3283,69%, а максимальна просадка (Max Drawdown) — лише 2,07%. Перспективи подальших досліджень включають: розробку інтегрованої методології управління ризиками на базі ML-моделей; впровадження механізмів раннього закриття позицій (зокрема, Trailing SL); та вдосконалення СППР для компенсації деградації моделі.

Ключові слова: алготрейдинг, FOREX, машинне навчання, прогнозування, класифікація, LSTM.

Recived 25.08.2025