

Method evolution and comparison in asset price prediction: a review from traditional statistical models to deep learning

Yusum Ng

School of Economics, Sichuan University, Chengdu, China

NGYUSUM95zz@outlook.com

Abstract. Asset price prediction is one of the most challenging research topics in finance. The inherent nonlinearity, low signal-to-noise ratio, and non-stationarity of Financial Time Series (FTS) make it difficult for traditional time series models to perform adequately. This paper reviews the evolution from traditional statistical models to Machine Learning (ML) and Deep Learning (DL), with a focus on comparing their performance in predictive accuracy, interpretability and robustness. Traditional models offer good interpretability and computational efficiency but are constrained by linear assumptions. ML models do not significantly outperform traditional methods, but exhibit conditional advantages that depend on asset characteristics and market conditions. DL models can capture complex temporal dependencies, but face challenges such as poor interpretability, overfitting and deployment difficulties. Based on current research, this review concludes that no universally optimal model exists, but only relatively suitable models exist for specific tasks, asset characteristics, and market environments. The future direction lies not in more complex architectures, but in reshaping prediction tools into interpretable decision support systems, which requires simultaneous advances in model architecture innovation, interpretability methods, and standardized evaluation frameworks.

Keywords: asset price prediction, financial forecasting, machine learning, deep learning, time series

1. Introduction

Financial markets generate massive amounts of data daily. The difficulty of asset price prediction stems from three inherent characteristics of financial data: nonlinearity, low signal-to-noise ratio, and non-stationarity.

Studies show that high-frequency financial data face issues such as low signal-to-noise ratio, asynchronous data, intraday seasonality, and imbalanced data. In addition, artificial intelligence, especially Deep Learning (DL), is increasingly applied to high-frequency data analysis, with datasets becoming more granular. However, this does not necessarily translate into improved predictive accuracy, as complex black-box models may overfit the data and thus underperform [1].

This dilemma has driven methodological evolution. Traditional time series models often struggle to capture nonlinear relationships, high volatility, and complex temporal dependencies in financial data. Subsequently, Machine Learning (ML) methods could significantly improve predictive accuracy in some cases. In recent years, DL has gone further, enabling automatic feature extraction and capturing complex temporal

dependencies. Standalone DL models such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) have been shown to outperform ML models like SVM and random forest [2]. However, this leap in predictive accuracy has come at the near-complete loss of interpretability. Against this backdrop, hybrid models attempt to strike a balance.

In recent years, several studies have attempted to outline this evolution. Giantsidi and Tarantola show that RNN-based models, especially LSTM, dominate financial forecasting research [3]. Vancsura et al. examine the application of AI and ML in financial forecasting, noting that neural networks outperform traditional statistical methods in capturing nonlinear and high-dimensional dynamics [4]. Lin and Lobo Marques further confirm that integrating more information sources and hybrid techniques improves predictive performance [5]. However, this does not mean that more complex models are always better, overfitting risk and the interpretability of black-box models remain major challenges [6].

Based on existing research, this paper systematically reviews the evolution from traditional statistical models to ML and DL in asset price prediction. It focuses on comparing methods in predictive accuracy, interpretability, and robustness, as well as their performance across different asset classes. On this basis, it discusses current challenges and future directions, aiming to provide an integrated reference for model selection in asset price prediction.

2. The intrinsic difficulties of Financial Time Series (FTS) forecasting and traditional prediction tools

2.1. Core challenges of FTS forecasting

The difficulty of FTS forecasting stems from three interrelated data characteristics: nonlinearity, low signal-to-noise ratio, and non-stationarity. These three are not independent but mutually embedded, constituting systemic obstacles for prediction models.

For asset price prediction, nonlinearity means that asset prices do not respond to information in a constant manner but vary with market states and shock directions. Tang et al. point out that complex and noisy real-time series data cannot be reflected by parametric analytical equations, because the dynamic equations of time series data are too complex [7]. The presence of nonlinear deterministic dynamics may be obscured by strong noise in the time series. Noise not only directly constrains predictive accuracy, but may also cause models to mistake random fluctuations for valid signals, leading to overfitting.

In their empirical analysis of daily return data from three financial indicators, Small and Tse found that all three time series are significantly distinguishable from linear noise or conditional heteroskedasticity models, exhibiting detectable deterministic nonlinear structures, and concluded that linear models can only provide an approximation of the underlying dynamics [8].

Non-stationarity may be the most fundamental challenge. Non-stationarity means that the statistical properties of a time series change with market conditions, which exposes any model trained on historical data to the risk of poor performance. The system generating FTS should be understood as a noise-driven nonlinear dynamical system that is either not in equilibrium or undergoing non-stationarity [8].

2.2. Traditional time series models: the cases of ARIMA and GARCH

ARIMA and GARCH are selected as traditional time series models because they are often used in nested form, constituting a typical benchmark in traditional econometrics that balances point prediction and dynamic risk characterization. Their core advantage is transparency, which many methods lack. The parameters of ARIMA

and GARCH have clear statistical meanings, and their model structures are well-defined, making them relatively interpretable.

ARIMA models have been widely applied in stock and foreign exchange price prediction, with the advantage that their coefficients have clear economic interpretations, suitable for understanding trends and cyclical changes. However, ARIMA models still have several limitations in FTS forecasting. First, ARIMA models assume that the differenced series is weakly stationary, and when this assumption is violated due to structural breaks or time-varying variance, prediction bias may arise. Second, ARIMA models typically assume normally distributed errors and cannot model the fat-tailed and leptokurtic distributions commonly observed in FTS.

The rise of GARCH models marked a major breakthrough in financial market volatility forecasting. GARCH models were developed by Bollerslev in 1986 based on the ARCH model and contribute to asset price prediction mainly by capturing the volatility clustering commonly observed in financial asset returns. GARCH models focus on the predictable dynamic structure of conditional variance, a feature that gives them broad application in risk management and option pricing.

However, traditional GARCH models still struggle with complex nonlinear market behavior and the leverage effect. To address these limitations, researchers have rapidly developed multiple extensions, including Nelson's EGARCH, Ding et al.'s PGARCH, and Zakoian's TGARCH, each characterizing market behavior from different angles [9].

Notably, with the help of R and mature software packages such as `forecast` and `rugarch`, the implementation and interpretation of ARIMA and GARCH models have become highly convenient. For example, ARIMA methods can make FTS stationary through appropriate differencing, though determining the optimal differencing order requires considerable time and skill to preserve important information in FTS. The `auto.arima()` function provided by the `forecast` package can automatically determine the appropriate time series model and order selection, making ARIMA predictions applicable to large numbers of series without manual intervention [10]. The `rugarch` package provides a unified framework for volatility modeling, supporting flexible configuration of various GARCH variants and enabling researchers to efficiently conduct model comparison and selection across different specifications.

The open-source ecosystem of R has not only lowered the barrier to applying traditional statistical models, but also sustained their vitality in the ML-dominated era via its transparency and community activity.

2.3. Monte Carlo simulation

Monte Carlo (MC) simulation can approximate the true distribution of financial assets by generating a large number of possible price paths. Its core value for financial forecasting lies in providing interval predictions rather than single-point predictions, thereby systematically characterizing uncertainty.

Singh et al. developed a multiple forecasting framework based on wavelet denoising and the ARIMA model [11]. Their results show that forecasts obtained through MC simulation using an integrated wavelet-denoising-ARIMA model become more accurate as the number of simulations increases, compared with applying the ARIMA model alone to noisy stock price series [11]. This finding conveys two messages. First, traditional ARIMA models still have room for improvement in financial forecasting, but this relies on preprocessing and simulation. Second, even within the traditional methodological framework, combining probabilistic methods represented by MC simulation can extend point predictions into full predictive distributions, thereby more comprehensively characterizing forecast uncertainty.

However, the quality of MC simulation is highly dependent on the underlying model specification. It cannot fundamentally resolve the model bias problem, because simulation merely transforms input

assumptions into output distributions, rather than validating the assumptions themselves. Nevertheless, the probabilistic thinking of replacing point estimates with distributions and generating predictive distributions through simulation provides methodological groundwork for the development of subsequent uncertainty quantification methods.

3. Machine learning

3.1. Kernel methods and nonlinear decision boundaries

Traditional time series models face systemic limitations in asset price prediction, and the introduction of ML provides a new path to overcome these limitations. Support Vector Machines (SVMs) were once widely used and considered among the better-performing algorithms. SVM is considered superior to traditional econometric models in volatility prediction. By mapping raw features related to asset prices into high-dimensional space through kernel methods, it makes originally linearly inseparable problems linearly separable and can easily handle high-dimensional data.

Ballings et al. took stock price direction as the prediction target, benchmarked seven methods, and used the Area under the Receiver Operating Characteristic Curve (AUC) as the metric. The results show that SVM ranked second, behind only Random forest (RF), indicating that SVM remains quite competitive in asset price prediction [12]. However, while kernel methods open up a nonlinear space, they also bring new problems: predictive power is limited by the prior hypothesis space defined by the kernel function and its parameters, and prediction results are difficult to trace back to specific price-driving factors.

3.2. Tree ensemble models and black-boxing

In asset price prediction, bagging and boosting are the two most widely applied ensemble strategies. When decision trees are used as base learners, they correspond to two representative tree ensemble models: RF and XGBoost. RF was proposed by Breiman in 2001. It employs bagging to integrate multiple trees in parallel to reduce the overfitting risk of a single tree, and is generally more robust in high-noise environments. XGBoost was proposed by Chen and Guestrin in 2016. It employs a boosting strategy to grow decision trees sequentially, and subsequently optimizes the overall prediction objective by fitting residuals or gradients, often achieving higher accuracy in scenarios with clearer signals.

The essential difference between the two strategies is that bagging mainly reduces variance, keeping the model stable in noisy environments, boosting mainly reduces bias, allowing the model to approximate the true function in signal-clear scenarios. This difference determines their division of labor in asset price prediction: the former suits high-noise markets, while the latter tends to be more advantageous in markets with stronger structural patterns. The stock market has a low signal-to-noise ratio, and RF's variance control mechanism helps prevent the model from capturing noise, while futures and foreign exchange markets are driven by relatively clear macroeconomic and arbitrage logic, and Boosting's continuous bias correction can more precisely approximate their intrinsic patterns.

The improvement in accuracy is accompanied by a continuous degradation of interpretability. SVM's kernel function transforms raw features into linear combinations in high-dimensional space, although difficult to reduce to specific economic variables, the support vectors at least provide a certain degree of decision traceability. The tree ensemble methods represented by RF and XGBoost, by contrast, disperse decision logic across a large ensemble of decision trees, making the derivation path of a single prediction almost untraceable.

3.3. Contributions and limitations of machine learning

The contribution of ML methods lies in that they break through the limitations of linear assumptions, automatically learn complex nonlinear patterns from data, and significantly improve predictive accuracy across multiple asset classes. However, the advantage of ML models in predictive accuracy is conditional, and the key lies in their ability to capture nonlinear structures. This breakthrough is particularly prominent in two scenarios. First, in high-dimensional feature spaces, where predictive information comes from a large number of technical indicators, industry variables, or macroeconomic indicators, automatic feature selection enables adaptive handling of such inputs. Second, in markets with stronger trends, boosting-type models may play a greater role.

However, ML has not fully resolved the core difficulties of financial forecasting. Vishwas et al. [6] pointed out future directions for interpretability optimization in asset price prediction, especially for ML and DL. First, adopt SHapley Additive exPlanations (SHAP) or Local Interpretable Model-Agnostic Explanations (LIME) to enhance model transparency and interpretability, and highlight the contribution of individual features. Second, introduce Attention Mechanisms (AMs) to identify key drivers of price movements. Third, combine causal inference with AI models, so that predictions not only rely on statistical correlations but also reflect the intrinsic drivers of market movements [6]. However, tools such as SHAP and LIME, while providing some degree of explanation at the feature contribution level, cannot fully reconstruct the model's decision logic.

4. Deep learning

4.1. Recurrent neural networks and their variants

DL further approaches the limit of predictive performance on the basis of ML. The Recurrent Neural Network (RNN) is a network structure designed for sequential data, but in actual training, gradients decay over time steps during backpropagation. The LSTM proposed by Hochreiter and Schmidhuber in 1997 solved this problem by introducing gating mechanisms and a cell state [13]. This addressed the difficulty of the RNN in capturing long-term dependencies such as market trends and cycles.

RNN, LSTM, and the Gated Recurrent Unit (GRU) are designed to handle time series data and have been shown to outperform traditional time series models when future events depend on a series of prior events [14]. LSTM is highly applicable to financial forecasting. Its gating mechanism effectively captures long-term dependencies in asset prices, but the cost is structural complexity and high computational overhead for training and inference. By contrast, GRU is a simplified version of LSTM and reduces computational cost at comparable performance, and may be more suitable for scenarios requiring simultaneous modeling of multiple assets. In addition, some studies suggest that simpler recurrent architectures may be more suitable for stock prediction in emerging markets [15]. Overall, RNN and its variants have improved predictive accuracy and robustness in asset price prediction.

4.2. Evolution of DL hybrid architectures

RNN, LSTM and GRU have improved predictive accuracy in asset price prediction, but the interpretability problem has not been alleviated. Subsequent progress has mainly followed two paths: first, further pushing accuracy through hybrid architectures, second, attempting to provide visual explanations for the decision logic of deep models through methods such as AMs.

Hybrid models can leverage the capabilities and advantages of various neural network architectures. Taking CNN-LSTM-AM as an example, CNN can extract local features and patterns from the input FTS, which helps

model short-term fluctuations, LSTM excels at capturing long-term dependencies such as market trends and cycles, AMs may further enhance the model's ability to focus on relevant time periods and features by dynamically attending to different parts of the input sequence at different prediction steps, thereby enabling it to effectively respond to market fluctuations [2]. However, hybrid models are more complex than standalone models, and their training cost and inference time are usually higher, which imposes certain constraints on real-time trading scenarios.

Clearly, complex hybrid models have again pushed up predictive accuracy, but still cannot solve the black-box problem. The Transformer architecture abandons recurrence and adopts self-attention mechanisms, allowing it to model dependencies across the entire sequence, and has received widespread attention in financial forecasting in recent years. Vancsura et al. [4] consider that the Transformer architecture has brought "revolutionary" development to time series forecasting, especially the Temporal Fusion Transformer (TFT), which delivers strong predictive performance and emphasizes interpretability, an important concern in regulated financial environments [4]. However, the Transformer architecture itself is still regarded as a black-box model and cannot provide direct interpretability. In addition, Hwang et al. point out that the Transformer has high data demands, which poses a considerable challenge because financial datasets are usually short, noisy, and costly to acquire [16].

Nevertheless, the black-box nature of deep learning remains a persistent concern in high-stakes financial environments. Although methods such as SHAP, LIME and attention mechanisms offer partial interpretability, achieving high predictive accuracy while maintaining transparency continues to pose a significant challenge [17].

5. Systematic comparison

5.1. Cross-dimensional comparison

In terms of predictive accuracy, some studies aim to find empirical evidence that algorithm performance is superior to traditional econometric models. However, the conclusion is that there is no definitive evidence that ML always outperforms traditional econometric models [18].

The three types of methods show a decreasing trend in interpretability. Traditional statistical models have fully transparent structures, with a direct correspondence between parameters and input variables. ML models do not have direct parametric interpretability, but tools such as SHAP can provide explanations of individual input features' contributions to predictions. DL models have the highest complexity and opaque internal decision mechanisms, and explanation tools can only provide approximate understanding.

The robustness of traditional statistical models is fundamentally constrained by their full-sample fixed-parameter estimation approach. Under structural breaks, ARIMA's estimated coefficients cannot reflect the data characteristics of the new regime, causing mean predictions to deviate significantly from reality. Although standard GARCH can capture short-term volatility shocks, its long-term volatility forecasts are constrained by a fixed long-run mean assumption and are difficult to adjust after volatility regime shifts. Vancsura et al. argue that ML in financial forecasting lacks research on temporal robustness and stability across different market environments and rarely applies time-dependent validation techniques, making it impossible to determine whether model performance remains stable over the long term or across different market cycles [4]. Giantsidi and Tarantola point out that DL currently also lacks robustness evaluation: few models are tested under adverse market conditions or subjected to stress scenario testing, which raises doubts about model robustness in real-world environments, especially amid regime changes and frequent extreme events. This problem is further exacerbated by the lack of standardized evaluation protocols, which continues to impede robust model

assessment [3]. At the same time, some researchers point out that there is a persistent gap between predictive accuracy and actual trading profitability, and that markets are adversarial and adaptive, with discovered market patterns decaying rapidly, which challenges the long-term effectiveness of any fixed model [16, 17].

In practice, most studies rely on statistical error metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), or on directional accuracy. While these measures are useful for comparing point-forecast performance, they do not capture economic value. Metrics that incorporate risk-adjusted returns (for example Sharpe ratios) or account for transaction costs are rarely reported. Moreover, the sensitivity of percentage-based metrics such as MAPE to high volatility remains under-examined, particularly in cryptocurrency markets [3, 4]. As a result, improvements in conventional statistical metrics frequently fail to translate into robust trading performance once realistic frictions and regime shifts are considered [4, 16].

5.2. Model selection based on asset characteristics

The comparison shows that the three types of methods each have strengths and weaknesses in accuracy, interpretability, and robustness, and no single model dominates across all dimensions simultaneously. Assets differ in their data and market characteristics, which impose different requirements on models. This indicates that model selection must return to specific asset characteristics and prediction contexts.

In stock prediction, LSTM remains the primary benchmark model, but empirical results increasingly favor new architectures, such as GRU, especially when they are tailored to specific market conditions or combined with technical indicators and sentiment features. Although many hybrid models outperform these DL benchmark models, these advantages usually come with a trade-off in interpretability [3]. ML methods have not been completely replaced, stock prediction often involves a large number of technical indicators, industry variables, and macroeconomic indicators, and the automatic feature selection capability of tree ensemble models can effectively handle such high-dimensional inputs.

Compared with individual stocks, indices usually exhibit lower volatility and stronger trends and better reflect economic conditions. LSTM, CNN and GRU are the most commonly used standalone architectures in this field. Notably, hybrid models, especially those combining CNN and LSTM, perform exceptionally well in index prediction [3]. By integrating technical and sentiment features, hybrid models can improve predictive accuracy, but overfitting remains a major challenge.

The foreign exchange market is the largest financial market by nominal trading volume, with extremely high liquidity. CNN is limited in such prediction tasks because the foreign exchange market is driven by macroeconomic factors and has a more complex temporal dependency structure. By contrast, LSTM architectures and XGBoost, or hybrid models combining macroeconomic and technical indicators, perform better. In addition, for high-frequency, short-term prediction tasks, Transformers with time embeddings are a competitive choice [19].

Currently, research in bond prediction is relatively scarce, making it difficult to draw broad conclusions, but it can be observed that the work mainly revolves around LSTM-based hybrid models. Future research can enrich inputs with alternative data such as inflation expectations, employment data, and ESG-related indicators to better capture bonds' sensitivity to macroeconomic and policy changes [3]. Traditional time series models remain competitive in structurally stable asset classes like bonds, and ARIMA-type models have unique value in terms of transparency and interpretability in predicting bond yields, which are driven by macroeconomic cycles and have relatively clear linear structures.

Cryptocurrency prediction is the fastest-growing field, yet model selection follows a distinct pattern compared with traditional assets. Research indicates that standard LSTM and machine learning models remain

the most commonly used approaches [3]. Given the extreme volatility and frequent regime shifts that characterize cryptocurrency markets, many existing studies continue to rely primarily on static evaluation metrics and limited temporal validation, which may constrain the robustness of the reported findings [4]. Hybrid and ensemble methods, however, still offer considerable room for further development.

For assets with limited data availability, such as certain commodities and emerging market assets, ML may have advantages over DL because it requires less data and has more manageable overfitting risk.

6. Conclusion

Traditional statistical models have transparent structures, but their linear assumptions struggle to handle the nonlinearity, low signal-to-noise ratio, and non-stationarity of financial data. ML models break through the linear limitation and can capture nonlinear patterns in data, but interpretability declines accordingly. DL models further improve predictive accuracy, but their black-box nature poses an obstacle in financial regulatory environments. Hybrid models attempt to balance accuracy and interpretability, but the increase in model complexity has not fundamentally solved the problem.

The most accurate models often have the worst interpretability, and the most robust models are not necessarily the most accurate in specific scenarios. This pattern determines that no universally optimal model exists, there is only the most suitable choice for a specific task context. The stock market is dominated by noise, so variance control is prioritized, the foreign exchange market is characterized by trends, so sequence modeling is prioritized, the bond market is driven by macroeconomic factors, so feature dimensionality reduction precedes architectural innovation, and cryptocurrencies are characterized by extreme volatility, so model adaptation speed is more critical than complexity. The starting point for model selection should be an understanding of asset data characteristics and task objectives, rather than the pursuit of model complexity.

The future direction of financial forecasting lies not in continuing to stack more complex architectures, but in reshaping prediction tools into interpretable decision support systems. This transformation requires coordinated progress in three dimensions: model architecture innovation should embed interpretability by design rather than as an afterthought, interpretability methods need to go beyond feature importance visualization and be deeply integrated with financial domain knowledge and regulatory requirements, and a standardized evaluation framework should be established to incorporate temporal robustness, cross-asset generalization ability, and actual trading profitability into the evaluation system.

References

- [1] Zhang, L., & Hua, L. (2025). Major issues in high-frequency financial data analysis: A survey of solutions. *Mathematics*, 13(3), 347.
- [2] Chen, W., Hussain, W., Cauteruccio, F., & Zhang, X. (2024). Deep learning for financial time series prediction: A state-of-the-art review of standalone and hybrid model. *CMES: Computer Modeling in Engineering & Sciences*, 139(1), 187–224.
- [3] Giantsidi, S., & Tarantola, C. (2025). Deep learning for financial forecasting: A review of recent trends. *International Review of Economics and Finance*, 104, Article 104719.
- [4] Vancsura, L., Tatay, T., & Bareith, T. (2025). Navigating AI-driven financial forecasting: A systematic review of current status and critical research gaps. *Forecasting*, 7(3), 36.
- [5] Lin, C. Y., & Lobo Marques, J. A. (2024). Stock market prediction using artificial intelligence: A systematic review of systematic reviews. *Social Sciences & Humanities Open*, 9, Article 100864.

- [6] Vishwas, H. N., Usha, B. A., & Sangeetha, K. N. (2025). Artificial intelligence-based stock market price prediction, a review. *Journal of Mathematical Models in Engineering*, 11, 204.
- [7] Tang, Y., Song, Z., Ding, W., Jia, W., Huang, L., & Wang, Y. (2022). A survey on machine learning models for financial time series forecasting. *Neurocomputing*, 512, 363–380.
- [8] Small, M., & Tse, C. K. (2003). Determinism in financial time series. *Studies in Nonlinear Dynamics and Econometrics*, 7(3), 1–28.
- [9] Qi, Y. (2025). Application of time series models in stock market volatility forecasting: A review of the literature in the past decade. *SHS Web of Conferences*, 225, Article 01014.
- [10] Hyndman, R. J., & Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software*, 27(3), 1–22.
- [11] Singh, S., Parmar, K. S., & Kumar, J. (2025). Development of multi-forecasting model using Monte Carlo simulation coupled with wavelet denoising-ARIMA model. *Mathematics and Computers in Simulation*, 230, 517–540.
- [12] Ballings, M., Van den Poel, D., Hespeels, N., Verdonck, T., & Van den Poel, D. (2015). Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications*, 42(20), 7046–7056.
- [13] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9, 1735–1780.
- [14] Song, H., & Choi, H. (2023). Forecasting stock market indices using the recurrent neural network based hybrid models: CNN-LSTM, GRU-CNN, and ensemble models. *Applied Sciences*, 13, 4644.
- [15] Nguyen, T. H., & Bui, T. K. (2026). GRU outperforms LSTM for Vietnamese stock market prediction: A comprehensive statistical comparison. In *2026 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems (ICETISIS)* (pp. 16–20). Manama, Bahrain.
- [16] Hwang, Y., Lee, Y., Kim, S., & Park, J. (2025). Deep learning in asset management: Architectures, applications, and challenges. *Journal of Portfolio Management*, 52(2), 233–271.
- [17] Noor Mohammadzadeh Maleki, D., Abbaskhani, S., Farhadi, A., Rahmani, A. M., & Hosseini, S. (2026). Artificial intelligence methods for financial market prediction: A systematic review. *Computers & Electrical Engineering*, 134, Article 111110.
- [18] Pérez-Pons, M. E., Parra-Dominguez, J., Corchado, J. M., & Herrera-Viedma, E. (2021). Machine learning and traditional econometric models: A systematic mapping study. *Journal of Artificial Intelligence and Soft Computing Research*, 12(2), 79–100.
- [19] Fischer, T., Sterling, M., & Lessmann, S. (2024). Fx-spot predictions with state-of-the-art transformer and time embeddings. *Expert Systems with Applications*, 249, Article 123538.