Reliable Stock Prediction: Data, Models, Testing

Guanglin Xie *

Guanghua Cambridge International School, Shanghai, China

* Corresponding Author Email: Manningharrison484@gmail.com

Abstract. In recent years, deep learning and large language models have entered almost every discussion on stock price prediction. Many reported results look strong on paper, but often rely on clean data, cheap trading, and generous assumptions that rarely hold in real markets. This review looks at studies from 2020–2025 through three practical lenses. First, data and task design: how prices, order books, and news are collected, filtered, labeled, and aligned with the information actually available at decision time. Second, models and multimodal methods: long-horizon forecasters, order-book based models, and text–market fusion schemes, all compared against simple but competitive baselines instead of strawman benchmarks. Third, evaluation and implementation: temporal cross-validation, tests for backtest overfitting, explicit treatment of transaction costs and liquidity, and the engineering and compliance constraints of deployable systems. Taken together, these perspectives argue that credible stock prediction work depends less on one more novel architecture and more on transparent data pipelines, honest testing, and designs that could survive live trading.

Keywords: stock prediction; time series; multimodal; backtest overfitting; reproducibility; execution costs.

1. Introduction

Stock price prediction not only has academic exploration significance but also directly connects to practical applications. It spans time series, financial text processing, and quantitative engineering. Between 2020 and 2025, the research focusses gradually shifted from "superimposing more complex structures" to "seeking stability under reasonable priors and strong baselines", and the evaluation approach also changed from simply pursuing error convergence to emphasizing out-of-sample reliability and tradability. The author believes that the key to understanding the real progress at this stage lies in focusing on the process and evidence: whether the data is aligned by event time, whether the segmentation avoids leakage, whether the significance has been corrected, and whether the experiments can be reproduced by third parties.

Based on this, this article focuses on several representative routes that can support the conclusion. First, explain the data and task Settings, and then discuss the development context of the model. Then, sort out the key points of integrating the text with market conditions, and provide practical approaches for evaluation and reproduction. Finally, in the section on challenges and prospects, the author of this article presents his judgments and suggestions. The issue of concern in this article is very specific: under the circumstances of limited samples, costs that cannot be ignored, and the market being able to switch at any time, how to conduct reliable, comparable, and reproducible research and transform statistical significance into executable strategies.

2. Data Sources and Prediction Tasks

Data and tasks are the foundation of the entire research. Price data (OHLCV) is the most common, and it is usually necessary to construct derivative quantities such as rolling mean, volatility and momentum on this basis. The text data is sourced from news, announcements and financial reports. It is necessary to use language models that have been continuously pre-trained in the financial field (such as FinBERT [1]) to encode the sentences into vectors, and then align them according to the target and time. Microstructure data (limit order book, LOB) provides factors such as spreads, depth, queues, and order flow imbalance, which are directly related to tradability and impact costs. The

author prefers to manage the three types of data in layers so as to gradually and incrementally test their contribution to the final performance in the experiment.

Time consistency is the first principle. Any standardization or dimensionality reduction steps should be fitted and frozen within the training window; otherwise, future information may be brought in, leading to "seemingly excellent" backtesting. For cross-market research, time zones and trading calendars should be unified. For historical data, recalculation and revision may occur. So, it is necessary to retain the data version and acquisition time to ensure that the conclusion is based on the information available "at that time". The rules and scripts for suspension, price adjusted for corporate actions (ex dividend or split) and handling of outliers still need to be made public so that readers can fully reproduce them.

A common oversight lies in making a one-time standardization of the entire time period and then dividing the training and testing into sections. In this way, the statistics from the testing period will be leaked into the training, making any model "look smarter". This study suggests standardizing through a rolling or extended window, and recording the timestamp and parameters of each fit in the log. If necessary, write them into the audit file for review and teaching.

For task setting, common practices include regression, direction classification and strategy output. Regression is oriented towards future earnings or prices. Classification aims to determine rises and falls and can set thresholds of the same magnitude as the spread. The strategy output directly provides the target position or order placement instructions. This study argues that the definition of labels must be linked to the executable price and strictly aligned with the features in terms of time. For instance, when the announcement is released at 10:00, the research can set an executable window between 10:05 and 10:30. Training and evaluation only use features that can be obtained before 10:00. For indicators, statistical indicators (such as MAE, MSE, AUC, etc.) and economic indicators (including cost-benefit, Sharpe, drawdown) should be reported in parallel and presented in segments at different market stages as much as possible to avoid misinterpreting phased advantages as long-term capabilities.

In feature engineering, this study suggests using logarithmic returns to mitigate scale differences; Adopt a gentle tailing for outliers rather than large-scale deletion. When learning across assets, the rolling z-score is adopted for cross-sectional standardization, while retaining industry or scale grouping to reduce structural noise. For intraday research, the time point and trading calendar effect can be incorporated to capture institutional patterns. Also, if target coding, aggregated statistics, or PCA are used, individual fitting should be carried out within each training window to avoid leakage caused by "global fitting".

3. Model Development (2020-2025)

Autoformer decomposed and autocorrelated stable long sequences by "trend-season" [2]; Zhou et al. proposed FEDformer implemented sparse attention in the frequency domain and strengthened global components [3]; Zeng et al. introduced DLinear established a strong baseline using low-complexity linear decomposition [4]; Nie et al. proposed PatchTST integrated cross-scale information through shard tokenization [5]; TimeGPT-1 [6] and Chronos [7] provide a unified risk interface with cross-domain pre-training and probabilistic output. Based on this, the author of this article believes that the combination of "prior + baseline + representation + basic model" is more reliable than a single structure.

The development of models presents a situation where "structural prior - strong baseline - representation modification - basic model" advances in parallel. Autoformer [2] and FEDformer [3] incorporate decomposable structures such as trends and seasons into the network and stabilize long sequence modeling through autocorrelation or frequency-domain sparsification. In this paper's view, the value of such methods lies in the constraints that are in line with the characteristics of the sequence. They can suppress overfitting when the sample size is limited, rather than simply increasing the number of layers and parameters.

The importance of a strong baseline has been repeatedly proven. DLinear characterizes trends and seasons in a linear way [4], while TSMixer uses alternating multi-layer perceptrons to blend in the "time-feature" two-dimensional dimension. Both perform stably on multiple benchmarks. They have a simple structure, low latency and are easy to deploy, and thus are often used as the first choice for online strategies. Based on this, the contributions of complex structures can be more clearly identified within the same evaluation framework.

Traditional methods are not completely without merit. In short-term, low-dimensional or low-signal-noise scenarios, linear models and tree models are often more stable and can provide clear feature importance and sensitivity analysis. The author of this article prefers to take them as the starting point in the reproduction experiment, and then gradually introduce the DLinear [4], TSMixer and Transformer series to measure the "true gain brought by complexity".

For representation transformation, PatchTST splits long sequences into segments as tokens [5], deepening the integration of cross-scale information. iTransformer reverses the attention object and uses the "variable dimension" as a token to depict the cross-sectional relationship among multiple assets. Time series basic models (TimeGPT [6], Chronos [7]) attempt cross-domain pre-training and directly output distributions or quantiles, thereby making them easier to couple with risk metrics such as VaR and ES. The author of this article reminds us that the noise structures in different markets vary greatly, and the migration of the basic model still needs to be gradually verified under strict segmentation.

At the micro level, DeepLOB directly learns short-term directions from the order book through convolutional and recurrent networks [8]. The reinforcement learning approach models transactions as cost-constrained sequential decision-making problems, absorbing both price and text signals simultaneously in the state. The common goal of the two routes is to narrow the distance between "signal - execution". Under different frequencies and market conditions, they can complement each other. To avoid overly aggressive strategies, the author prefers to incorporate hard constraints such as risk budgeting and transaction costs, and use a rebalancing cooling-off period to control turnover.

About the training paradigm, a trade-off needs to be made between "direct multi-step prediction" and "recursive rolling". The former outputs goals of multiple timeframes at one time, which is stable but has a relatively high training cost. The latter is more efficient, but the error spreads faster. This study suggests comparing the economic effects of the two schemes under a unified cost-inclusive evaluation rather than merely comparing statistical errors. Regularization is as crucial as early stopping: Weight decay, dropout, and random seed fixation are not minor issues but rather preconditions for ensuring the replicability of conclusions.

4. Text and Market Multimodal Fusion

FinBERT provides financial-domain language representations [9]; FinGPT opens an end-to-end pipeline for financial large language models [10]. Stock-UniBERT [11] and FinBERT-LSTM [12] report improvements of news-driven methods in directional prediction. Fazlija and Harder [13] and Cristescu et al. [14] find significant relations between media sentiment and returns at broader horizons.

Integrating text and market conditions is important because many price jumps are triggered by information. For specific implementation, it is necessary to first perform duplicate news removal and entity disambiguation, and then encode "emotion - entity - event - time" into vectors and align them with the price characteristics within a unified window. If these details are ignored, even if the model has an advantage in statistical indicators, it may still be difficult to translate into executable benefits. For instance, if the release time of the financial report summary is different from that of the full version and no specific time point is specified, backtesting may overestimate the "response speed" of the strategy.

The core of moving from "effective information" to "feasible transactions" lies in cost and time. A more rigorous approach is to explicitly incorporate commissions, spreads and slippage into the training objectives, or to use cost-sensitive learning to punish high turnover. During the evaluation

stage, an executable window is set based on the event time, and information freezing is established to avoid misjudging the reporting lag as "early prediction". This study prefers to adopt the "default settings" of incorporating these constraints into the research, thereby making the comparison fairer.

For engineering implementation, the two-stage pipeline of recall and reranking can enhance robustness while controlling latency. That is, first, a lightweight model is used to screen out potential signals, then a relatively complex structure is used for re-scoring, and an intermediate representation is cached between the two segments to reduce the inference cost. In this article's view, this approach is also applicable to multi-asset cross-sectional expansion.

5. Evaluation and Reproducibility

Bailey, Borwein, Lopez de Prado and Zhu proposed backtest overfitting [15] probability (PBO) for identifying the "winner picking" risk; Bailey and Lopez de Prado proposed the Deflated Sharpe Ratio (DSR), which provides a more conservative significance judgment after considering the sample length and distribution pattern. The author suggests reporting both PBO and DSR simultaneously under the condition of consistent time segmentation.

Evaluation and reproducibility determine whether the conclusion can hold water. Ordinary K-fold cross-validation can disrupt temporal causality and is prone to leakage. Purged/Embargoed K-Fold suppresses this problem by purifying overlapping samples and setting an embargo period after verification; CPCV/CSCV further divides the time axis into multiple blocks and repeatedly validates them in a non-adjacent combination manner to obtain multiple sets of out-of-sample results. On this basis, the backtest overfitting probability (PBO) can be estimated to quantify the risk that "the best in training does not necessarily mean robust" [15].

Significance correction should not be ignored either. The return sequence is often non-normal and influenced by multiple experiments. Direct comparison with Sharpe is often overly optimistic. The Deflated Sharpe Ratio (DSR) takes into account factors such as sample length and skewness/kurtosis to provide a more conservative confidence judgment. This paper suggests providing the original Sharpe, DSR and the minimum significant sample length simultaneously. All gains and drawdowns are calculated under cost-inclusive conditions, along with turnover, holding periods and capacity sensitivity. For important conclusions, they should be repeatedly verified in independent resegmentation intervals.

To facilitate verification, the research should also provide reproducible experimental materials: data sources and versions, time zones and trading calendars, feature scripts and configuration files, random seeds and logs. Walk-forward backtesting (rolling or extended window retraining, fixed-frequency rebalancing) should also be clearly stated in the text. The author of this article advocates the use of a uniform template for chart presentation, including cost-benefit curves, stage performance, OOS index tables, and parameter sensitivity curves, allowing readers to intuitively compare the stability of different methods. For the manuscripts submitted for review, a "minimum reproduction package" should also be uploaded, which should at least include a data dictionary, feature scripts and main configuration files.

Execution and Micro: DeepLOB learns short-term directions directly from order books [8]; subsequent work models execution as cost-constrained sequential decisions solved via reinforcement learning in realistic limit order book simulations [16]. FinRL provides deep reinforcement learning environments and libraries for trading strategies [17], and FinRL-Meta offers standardized market environments and benchmarks, including ICAIF tasks [18]. Signals and execution should be evaluated within a unified, realistic environment rather than in isolation.

6. Challenges and Future Directions

Although the process is more standardized, difficulties still objectively exist. Non-stationary and institutional switching may change the revenue structure in a short period of time. Online fine-tuning,

meta-learning or invariant risk minimization are needed to improve the adaptation speed. In actual deployment, monitoring drift and setting up a fallback mechanism are necessary conditions for maintaining long-term performance.

The causal interpretation of text signals is also not easy. The rise in popularity does not necessarily translate into executable returns. This article prefers to combine event studies and instrumental variables and other methods to make more cautious inferences on a strictly aligned timeline. When the experimental results contradict intuition, priority should be given to checking whether the time alignment and sample segmentation are appropriate.

Also, there is still a lack of a universal differentiable mechanism for stably mapping the transaction probability and impact at the order book level to the weights and risk control at the portfolio level. Differentiable matching and high-fidelity simulation may offer a path, but both their data requirements and engineering complexity are not low. This study suggests first verifying in a small-scale sandbox environment and then gradually expanding the scope of funds and markets.

Migration between different markets is also often restricted. Due to the significant differences in systems and liquidity, cross-market generalization requires additional regularization and re-segmentation tests. The sensitivity curves of key parameters should be provided in the report, along with the failure scenarios, to avoid making excessive commitments regarding external capabilities.

Research ethics is as important as disclosure. If automated writing or machine-generated code is used, it should be truthfully stated in the method or appendix, and ensure that all expressions and citations are traceable.

7. Engineering Cost and Compliance

Knowledge distillation, pruning and caching can cut inference time and memory use. At the system level, set blacklists, trading windows and position limits. Keep a one-click audit pack (data version, code, configs, logs) to meet review and regulation needs. In practice, research and engineering should have balanced weight.

8. Conclusion

This review set out to ask a simple question: under what conditions do stock prediction models produce signals that could plausibly survive real trading, rather than only look persuasive in backtests? Looking across recent work from 2020 to 2025, three themes emerge.

First, data and task design decide the ceiling. The choice of price sources, order-book reconstruction, corporate action handling, survival bias, text collection, and label definition determines whether the learning problem is even well-posed. Studies that do not align features with the true information set, or that treat illiquid assets and zero-cost execution as harmless simplifications, provide at best an optimistic upper bound, not an implementable result.

Second, model advances are meaningful only when anchored to clear priors and strong baselines. Long-horizon transformers, order-book networks, and multimodal architectures drawing on FinBERT, FinGPT and related tools can exploit richer structure, but their contribution appears only when they are compared against transparent linear models, tree ensembles, and simple rules under identical conditions. The evidence so far suggests that stability comes less from architectural novelty and more from reasonable inductive biases, careful representation of time and cross-section, and disciplined regularization.

Third, evaluation, execution, and engineering must be treated as part of the method, not an afterthought. Time-aware cross-validation, tests for backtest overfitting, cost and liquidity modeling, realistic execution assumptions, and reproducible pipelines built on open environments such as FinRL and FinRL-Meta together form a minimal standard for credible claims. A model that cannot be reproduced, audited, or mapped to a feasible trading protocol should not be counted as progress.

Taken together, these observations point to a process-first benchmark for future work. Robust stock prediction research will be defined less by one more state-of-the-art curve, and more by transparent data pipelines, conservative evaluation, and implementable designs that acknowledge how real markets trade.

References

[1] Huang A H, Wang H, Yang Y. FinBERT: A Large Language Model for Extracting Information from Financial Text. Contemporary Accounting Research, 2023, 40(2): 806-841.

[2] Wu H, Xu J, Wang J, Long M. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. Advances in Neural Information Processing Systems (NeurIPS), 2021.

[3] Zhou T, Ma Z, Wen Q, Wang X, Sun L, Jin R. FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting. International Conference on Machine Learning (ICML), 2022: 27268-27286.

[4] Zeng A, Chen M, Zhang L, Xu Q. Are Transformers Effective for Time Series Forecasting? AAAI Conference on Artificial Intelligence, 2023: 11121-11128.

[5] Nie Y, Nguyen N H, Kalagnanam J. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. 2023.

[6] Garza A, Challu C, Mergenthaler-Canseco M. TimeGPT-1. 2023.

[7] Ansari A F, Stella L, Türkmen C, Zhang X, Mercado P, Hassani H, V den Broeck G. Chronos: Learning the Language of Time Series. 2024.

[8] Zhang Z, Zohren S, Roberts S J. DeepLOB: Deep Convolutional Neural Networks for Limit Order Books. IEEE Trans. Signal Processing, 2019, 67(11): 3001-3012.

[9] Araci D. FinBERT: Financial Sentiment Analysis with Pre-Trained Language Models. 2019.

[10] Yang H, Liu X, Zhou Z, et al. FinGPT: Open-Source Financial Large Language Models. 2023.

[11] Man X, Lin J, Yang Y. Stock-UniBERT: A News-Based Composable Stock Forecasting System using Deep Neural Networks. 2020 IEEE 18th International Conference on Industrial Informatics (INDIN), 2020: 440-445.

[12] Gu W J, Zhong Y H, Li S Z, Wei C S, Dong L T, Sha Z M, Cheng X Q. Predicting Stock Prices with FinBERT-LSTM: Integrating News Sentiment Analysis. Proc. ICCBDC, 2024.

[13] Fazlija B, Harder P. Using Financial News Sentiment for Stock Price Prediction. Mathematics, 2022, 10(13).

[14] Cristescu M P, Nerişanu R A, Dumitru A M. Using Market News Sentiment Analysis for Stock Market Prediction. Mathematics, 2022, 10(22).

[15] Bailey D H, Borwein J M, López de Prado M, Zhu Q J. The Probability of Backtest Overfitting. Journal of Computational Finance, 2016, 20(5): 39-69.

[16] Karpe M, Fang J, Ma Z, Wang C. Multi-Agent Reinforcement Learning in a Realistic Limit Order Book Market Environment. ACM ICAIF, 2020.

[17] Liu X-Y, Yang H, Gao J, Wang C D. FinRL: Deep Reinforcement Learning Framework for Quantitative Finance. ACM Int. Conf. on AI in Finance (ICAIF), 2021.

[18] Liu X-Y, Xia Z, Rui J, Yang H, Zhu M, Wang C D, Zhang Z, et al. FinRL-Meta: Market-Environments Library for Data-Driven Financial Reinforcement Learning. NeurIPS Datasets and Benchmarks Track, 2022.