Proceedings of CONF-BPS 2026 Symposium: GenAI, Labour Markets, and the Economics of Human and Financial Capital
DOI: 10.54254/2754-1169/2026.LD31883

Empirical Research on Multi-factor Prediction and Algorithmic Trading Strategy Based on Transformer Model

Lehan Wu

Broad College of Business, Michigan State University, East Lansing, USA
wulehan@msu.edu

Abstract. The field of quantitative finance has undergone significant transformations due to the integration of artificial intelligence, particularly in the context of short-term stock return forecasting. Transformer models have demonstrated considerable aptitude in modeling intricate time-series dependencies; nevertheless, their efficacy in predicting the subsequent return of specific stocks, such as AAPL, remains a relatively unexplored domain. In this study, systematic collection of daily price, volume, and other fundamental information for AAPL was conducted from public data sources (e.g., Yahoo Finance), and a multi-factor dataset incorporating volatility, value, liquidity, momentum, yield, and sentiment factors was constructed. A transformer encoder architecture has been developed to capture temporal relationships among these factors and to generate 5-day return forecasts. Empirical results demonstrate that the Transformer exhibits lower forecast accuracy in this limited-sample single-asset setting in comparison with traditional machine-learning benchmarks. However, when model signals are translated into a unified soft-position long-only strategy with transaction costs, strategy-level performance differences become more pronounced, and the Transformer-based strategy achieve marginally higher risk-adjusted returns and total returns. The findings indicate that, while the guarantee of predictive superiority is not provided, Transformer-based factor forecasting may still generate economic value under disciplined execution and position scaling..

Keywords: Transformer model, Multi-factor prediction, Algorithmic trading, Deep learning, Stock price prediction

1. Introduction

Machine learning has been extensively applied in stock return forecasting, and numerous traditional models have demonstrated reasonable performance [1]. However, these models frequently encounter difficulties in accurately capturing the intricate temporal dependencies inherent in financial time-series data [2]. Conversely, the Transformer architecture has been engineered to model long-range sequential patterns with greater efficacy through its self-attention mechanism [3]. Recent studies have demonstrated the significant potential of Transformer-based models in various financial applications, particularly in stock price forecasting [4]. In this context, Transformer models have been shown to process long historical sequences and extract meaningful predictive signals that outperform classical approaches [5].

Despite this progress, the majority of existing research concentrates on multi-asset portfolios or cross-sectional prediction across extensive sets of stocks [6]. Few studies have explored the application of Transformers in forecasting the 5-day return of a single stock and generating corresponding trading strategies. This gap is significant because individual stock forecasting poses distinct challenges and is more susceptible to factor selection, noise, and market regimes [7]. Therefore, this study investigates whether a Transformer model can effectively predict the 5-day return of a single stock—specifically, AAPL, and develop a profitable trading strategy based on its forecasts.

To address this, several years of AAPL data were collected from Yahoo Finance, and a multi-factor dataset consisting of volatility, value, liquidity, momentum, yield, and sentiment indicators was constructed. This study compares the prediction and trading performance of the Transformer model against several benchmark models, including Random Forest, Ridge, and GRU(a parameter-efficient gated RNN). By integrating model predictions with systematic trading strategies, this study aims to contribute to the literature by filling an important gap in single-stock multi-factor forecasting using Transformer architectures. In addition, this study demonstrates their potential value in practical algorithmic trading.

2. Methodology

2.1. Data collection

To ensure the reliable computation of rolling-window technical indicators, an extended sample of AAPL data covering 2018–2024 was collected. The observations from 2018 to 2019 are utilized solely for the preparatory phase of factor construction, thereby mitigating the loss of early-sample data caused by lookback requirements. The initial modeling and evaluation period extends from January 2020 to December 2024.

The dataset comprises daily market variables, including open, high, low, close, and trading volume, as well as fundamental indicators such as the price-to-earnings (P/E) ratio, price-to-book (P/B) ratio, dividend yield, and selected quarterly financial measures (e.g., revenue and earnings growth). Fundamental variables are reported at lower frequencies; therefore, they are aligned to the daily frequency via forward-filling (carrying forward the most recently available observation) to match the daily input requirements of the model.

2.2. Factor construction

Following prior literature, this study begins with a multi-factor candidate set spanning categories commonly used in empirical asset pricing research, including Size, Value, Momentum, Growth, Liquidity, Volatility, Yield, Sentiment, and Quality. These factors capture different economic dimensions of stock behavior. For each factor category, specific measurable variables suitable for single-stock analysis are constructed. Factors with excessive missing are excluded to avoid heavy imputation.

Specifically, factors with a missing rate exceeding 30% within the modeling window are excluded after forward-fill alignment.

The Growth factor and Quality factor retrieved from Yahoo Finance remain sparsely available at daily frequency even after alignment, thus being excluded from the final feature set.

Following the empirical asset pricing literature, seven commonly used factor categories are ultimately adopted. The definitions of factors X_1 - X_7 are summarized in Table 1.

Table 1. Factor tab

Factor	Symbol	Factor Name	Description
X1	SIZE	Size Factor	Risk associated with company size.
X2	VALUE	Value Factor	Risk associated with book-to-market valuation.
X3	MOMENTUM	Momentum Factor	Risk associated with stock price momentum.
X4	LIQUIDITY	Liquidity Factor	Represents the stock's liquidity.
X5	VOLATILITY	Volatility Factor	Represents the stock's price volatility.
X6	YIELD	Yield Factor	Represents the dividend yield.
X7	SENTIMENT	Sentiment Factor	Measures the impact of investor sentiment.

2.3. Data cleaning and processing

Missing values arising from non-trading days or indicator computation windows are addressed through forward-fill procedures. Lag alignment is applied to ensure that no future information is inadvertently incorporated into the model, particularly when constructing 5-day returns and computing technical indicators.

Winsorization thresholds and z-score normalization parameters are fitted on the training set only and then applied to validation and test sets to prevent look-ahead bias.

The 5-day stock return is set as the prediction target. To mitigate the impact of extreme price movements, the raw return series is winsorized at the 1% and 99% quantiles:

$$y_t = \mathrm{clip}(r_{t+1}, q_{1\%}, q_{99\%}) \tag{1}$$

where r_{t+1} denotes the realized 5-day return, and q represents the quantile threshold.

All factor features are standardized using z-score normalization:

$$x'_i = \frac{x_i - \mu_x}{\sigma_x} \tag{2}$$

where μ_x and σ_x denote the time-series (training-sample) mean and standard deviation of each factor.

Missing values arising from indicator computation windows or quarterly fundamental updates are forward-filled to ensure temporal consistency.

2.4. Transformer model architecture

To model the complex temporal dependencies in the multi-factor time series, an encoder-only Transformer architecture is employed. Let the input sequence for day t be defined as:

$$X_t = \{x_{t-T+1}, x_{t-T+2}, \dots, x_t\} \tag{3}$$

where each $x_i \in R^d$ represents the vector of d standardized factor values, and T denotes the look-back window length. Each input vector is first passed through a linear projection layer to map it into a latent space of dimension d_{model} :

$$h_i = W_e x_i + b_e \tag{4}$$

A learnable positional encoding p_i is added to each embedded vector to convey temporal ordering information:

$$z_i = h_i + p_i \tag{5}$$

The sequence $Z=\{z_1,z_2,...,z_T\}$ is then processed by multiple Transformer encoder layers. Each encoder layer contains a multi-head self-attention mechanism. For a single attention head, the computation is given by:

$$\text{Attention}(Q,K,V) = \text{softmax}(\frac{QK^\top}{\sqrt{d_k}})V \tag{6}$$

Where $Q=ZW_Q$, $K=ZW_K$, and $V=ZW_V$, with W_Q , W_K , W_V being learnable parameter matrices. Outputs from all heads are concatenated:

$$\text{MultiHead}(Z)=\text{Concat}(\text{head}_1,...,\text{head}_H)W_O \tag{7}$$

Residual connections and layer normalization are applied to both the multi-head attention and feed-forward sub-layers:

$$Z_{attn}=\text{LayerNorm}(Z+ \text{MultiHead}(Z)) \tag{8}$$

$$Z_{ffn}= \text{LayerNorm}(Z_{attn}+ \text{FFN}(Z_{attn})) \tag{9}$$

The feed-forward network (FFN) consists of two linear transformations with a non-linear activation:

$$\text{FFN}(z)=W_2\sigma(W_1z+b_1)+b_2 \tag{10}$$

Stacking L such encoder layers yields the final contextual representation. The hidden state corresponding to the last time step, denoted $Z_T^{(L)}$, summarizes the relevant historical information and is passed through a fully connected output layer:

$$\widehat{r_{t+1}}= W_O Z_T^{(L)} + b_O \tag{11}$$

which produces the predicted 5-day return.

This architecture enables the model to learn nonlinear interactions among factors as well as long-range dependencies within the time series, making it particularly suitable for multi-factor stock return forecasting.

2.5. Training procedure

The datasets is chronologically divided into three non-overlapping subsets:

·Training set: January 2020 – December 2022

·Validation set: January 2023 – December 2023

·Test set: January 2024 – December 2024

This temporal split ensures that no future information is used during model training, thereby preventing look-ahead bias. The model is trained to forecast the 5-day return $\widehat{r_{t+1}}$ using the historical factor window X_t . The training objective is to minimize the mean squared error (MSE) loss:

$$L_{MSE}=\frac{1}{N}\sum_{i=1}^{1}\left(y_i-\widehat{y_i}\right)^2 \tag{12}$$

Where y_i denotes the winsorized 5-day return and $\widehat{y_i}$ is the model prediction.

The Adam optimizer with an adaptive learning rate is used, as it is well-adapted to addressing the non-stationarity and noisy gradients present in financial time-series data. Hyperparameters such as the learning rate, sequence length T , embedding dimension d_{model} , number of attention heads, number of encoder layers, and dropout rate are tuned through the validation set to avoid overfitting. A typical configuration includes:

·Learning Rate: 1×10^{-4}

·Batch Size: 32

·Sequence Length T : 60 trading days

·Encoder Layers: 3

·Attention Heads: 8

·Dropout: 0.1–0.2

To enhance generalization, several regularization techniques are applied. Dropout is used in both the multi-head attention and feed-forward layers. Weight decay constrains parameter growth, and early stopping is employed: training is terminated when the validation loss fails to improve for several epochs. This prevents overfitting to volatile market conditions.

Once trained, the model is evaluated on the test set using both predictive metrics (MSE, MAE, hit rate) and economic metrics derived from the trading strategies described in Section 3.5. The final model selected is the one that balances predictive accuracy and stability across the validation and test periods.

2.6. Trading strategy design

To evaluate the economic usefulness of model forecasts, a long-only trading strategy is constructed. Let $\widehat{r_t^{(5)}}$ denotes the model's forecast of the 5-day forward cumulative return, as formed using information that was available at time t. Rather than adopting an all-in/all-out rule, the continuous forecast is mapped to a smooth long-only portfolio weight to reduce turnover and improve stability. The specific allocation of the portfolio's assets to Apple Inc. (AAPL) is delineated as follows:

$$w_t=\sigma(k\bullet\widehat{r_t^{(5)}})=\frac{1}{1+\exp(-k\bullet\widehat{r_t^{(5)}})} \tag{13}$$

Where $w_t\in(0,1)$and the remaining $1-w_t$ is held in cash (assumed to earn zero return). The parameter $k>0$ controls signal sensitivity: larger k yields, more aggressive exposures (closer to 0/1), while smaller k produces smoother positions and lower turnover. The strategy is rebalanced daily based on the most recent forecast $\widehat{r_t^{(5)}}$. Although the predictive target is a 5-day return, daily rebalancing allows the model to update exposure as new information becomes avialable.

Strategy Returns:

Strategy returns are computed daily as:

$$R_{t+1}=w_t r_{t+1}-c\left|w_t-w_{t-1}\right| \tag{14}$$

Where r_{t+1} is the realized one-day return of AAPL from t to $t+1$, and c is the one-way proportional transaction cost (set to 10 bps in the baseline evaluation). The term $\left|w_t-w_{t-1}\right|$ measures turnover induced by rebalancing. Performance is evaluated using cumulative return, annualized return, annualized volatility, Sharpe ratio, and maximum drawdown, providing a comprehensive view of profitability and risk.

To incorporate realistic market frictions, a 10-basis-point (0.1%) transaction cost per trade is included in robustness checks.

Performance is evaluated using cumulative return, annualized return, Sharpe ratio, maximum drawdown, and directional hit rate, enabling a comprehensive understanding of both profitability and risk characteristics.

2.7. Evaluation metric

Model performance is assessed from both predictive and economic perspectives. Predictive performance is evaluated using the mean squared error (MSE), mean absolute error (MAE), and directional accuracy (hit rate).

The hit rate measures the frequency with which the model correctly predicts the sign of returns:

$$\text{Hit Rate} = \tfrac{1}{N}\sum_{t=1}^{N} \mathrm{I}(\widehat{r_t^{(5)}\bullet}\, \widehat{r_t^{(5)}}>0) \tag{15}$$

Where $\mathrm{I}()$ is the indicator function.

Economic Evaluation

·To evaluate the profitability and risk of the strategy, the following metrics are computed:

·Cumulative Return

·Annualized Return

·Annualized Volatility

·Maximum Drawdown

·Sharpe Ratio, defined as:

$$\text{Sharpe} = \frac{E\left[r_t^{\text{strategy}}\right]}{\sigma\left(r_t^{\text{strategy}}\right)} \tag{16}$$

where $E\left[r_t^{\text{strategy}}\right]$ is the average daily strategy return and $\sigma\left(r_t^{\text{strategy}}\right)$ is the standard deviation of returns.

These metrics jointly assess both forecast accuracy and practical trading performance, enabling a comprehensive comparison between Transformer-based strategies and benchmark models.

3. Research result

3.1. Model predictive performance comparative analysis

To systematically evaluate the effectiveness of different modeling paradigms in single-stock multi-factor forecasting, four predictive models are constructed and compared: a Transformer-based sequence model, a GRU (RNN) model, and two traditional machine learning baselines (Ridge regression and Random Forest). All models are trained and tested on the same factor datasets and evaluated under an identical chronological split (train: 2020–2022; validation: 2023; test: 2024). Model forecasts are subsequently converted into a unified trading rule using the soft-position long-only strategy. The sensitivity parameter $k=5$ is selected on the validation set and fixed for all models to ensure a fair comparison.

Predictive performance is summarized in Table 2. The empirical results provide a clear ranking of the modeling paradigms. The GRU model emerges as the strongest predictor, achieving the lowest test error (MSE = 0.000934; MAE = 0.024510) and the highest directional and correlation-based metrics (Hit Rate = 70.97%, IC = 0.4568, Rank IC = 0.5198). In comparison, traditional baselines maintain competitiveness: Ridge regression yielded a result of Rank IC = 0.3809, while Random Forest attained a result of Rank IC = 0.3269. Notably, the Transformer model demonstrates lower predictive accuracy in this single-asset context, as evidenced by metrics such as MSE (0.001018) and MAE (0.025868). Additionally, it exhibits a comparatively lower correlation with actual outcomes (IC = 0.3061; Rank IC = 0.2813). These findings indicate that, for a limited-sample, single-stock time series, recurrent architectures may exhibit superior data efficiency in capturing short- to medium-term temporal patterns when compared to a higher-capacity attention-based model.

Table 2. Comparative analysis of multi-model predictive performance

No.	Model	MSE	MAE	Hit Rate	IC(Pear)	IC(Spear)
1	GRU(RNN)	0.000934	0.024510	0.709677	0.456843	0.519771
2	Ridge	0.000983	0.024392	0.608871	0.382187	0.380861
3	Random Forest	0.000964	0.024433	0.596774	0.336225	0.326920
4	Transformer	0.001018	0.025868	0.564516	0.306118	0.281327

Table 3. Comparative analysis of multi-machine learning model strategies

No.	Model	k	Total Return	Sharpe Ratio	MDD	Turnover
1	Transformer	5	0.200322	1.666153	0.081707	0.006295
2	GRU(RNN)	5	0.194394	1.655417	0.079524	0.009143
3	Random Forest	5	0.192556	1.646223	0.080305	0.004904
4	Ridge	5	0.190100	1.646081	0.078780	0.003936

3.2. Model strategy performance comparative analysis

Strategy performance is summarized in Table 3, where all models are evaluated under the same soft-position long-only strategy with transaction costs of 10 basis points (bps). In contrast to the predictive ranking, trading outcomes exhibit substantially smaller dispersion across models. Following the implementation of the unified position-sizing overlay, all strategies exhibited comparable risk-adjusted performance, with Sharpe ratios tending to a value of 1.65 and maximum drawdowns reaching approximately 8%. Among the strategies, the Transformer-based strategy achieves the highest test Sharpe ratio (Sharpe = 1.666) and the highest total return (Total Return = 0.2003), with a controlled drawdown (MDD = 8.17%) and low turnover (0.0063). The GRU strategy performs consistently within the established framework (Sharpe = 1.655; Total Return = 0.1944; MDD = 7.95%), while Random Forest and Ridge deliver comparable Sharpe ratios (both ≈ 1.646) and analogous drawdowns. The findings reveal that while predictive metrics vary considerably across models, a conservative exposure-scaling mechanism can mitigate strategy-level discrepancies by smoothing signals and constraining aggressive position changes, resulting in trading performance that is largely comparable across models.

Figure 1 provides a visual illustration of the findings. The net value curves of the soft-position strategies remain closely aligned throughout the test period, indicating the stabilizing effect of position scaling under transaction costs. The Transformer curve ends slightly above the benchmarks, aligning with its modest superiority in the Sharpe ratio and terminal net value as documented in Table 3.

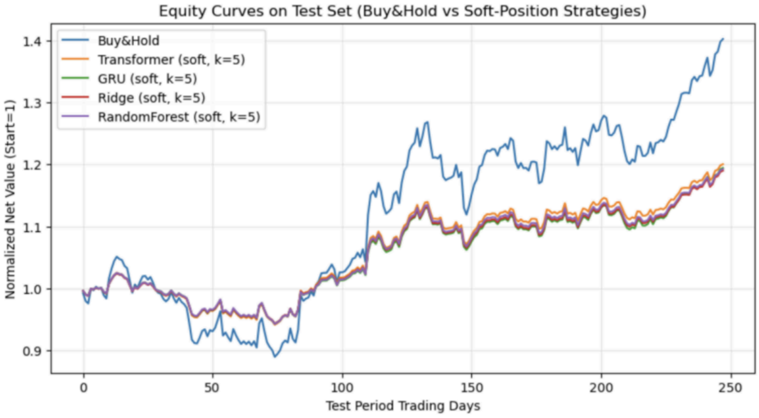

Figure 1. Equity curves on test set(buy & hold vs soft-position strategies)

4. Conclusion

This study investigates the effectiveness of a Transformer-based framework for multi-factor return forecasting and algorithmic trading in a single-stock setting. Using AAPL daily data from Yahoo Finance, a 7-factor dataset encompassing value, liquidity, volatility, and technical indicators is constructed. To mitigate missing values arising from rolling-window factor construction, factor computation begins in 2018, while model estimation and evaluation focus on 2020–2024 with a strict chronological split (train: 2020–2022; validation: 2023; test: 2024). The prediction target is the 5-day forward cumulative return y_{5d}, and forecasts are further translated into trading decisions to assess economic significance.

Benchmark comparisons indicate that model architecture influence predictive accuracy in a single-asset time series. On the 2024 test set, the GRU model achieves the strongest predictive performance (MSE = 0.000934; MAE = 0.024510; Hit Rate = 0.709677; IC = 0.456843; RankIC = 0.519771), outperforming Ridge regression and Random Forest. In contrast, the Transformer delivers weaker predictive metrics (MSE = 0.001018; MAE = 0.025868; Hit Rate = 0.564516; IC = 0.306118; RankIC = 0.281327). These findings suggest that under limited sample size and a single-stock context, recurrent architectures may be more data-efficient in extracting short- to medium-horizon temporal patterns than a higher-capacity attention-based model.

To evaluate economic value, a unified soft-position long-only strategy with transaction costs is implemented, and the sensitivity parameter k is selected exclusively on the validation set to avoid test-set tuning. The validation search selects $k=5$ for the Transformer, and the same k is fixed for all models in test evaluation to ensure a fair comparison. Under this disciplined execution framework (cost = 10 bps), strategy-level performance differences narrow substantially: the

Transformer attains the highest test Sharpe ratio (Sharpe = 1.666153) and total return (Total = 0.200322), with controlled downside risk (MDD = 0.081707) and low turnover (0.006295). The benchmark strategies are close, including GRU (Sharpe = 1.655417; Total = 0.194394; MDD = 0.079524), Random Forest (Sharpe = 1.646223; Total = 0.192556; MDD = 0.080305), and Ridge (Sharpe = 1.646081; Total = 0.190100; MDD = 0.078780). Overall, the results highlight that while predictive metrics differ clearly across models, conservative position sizing and trading frictions can compress performance dispersion at the strategy level, enabling the Transformer to remain competitive in risk-adjusted returns.

This study has several limitations. First, the results are based on a single stock and may not apply to all assets or scenarios. Second, factor coverage is limited to publicly accessible Yahoo Finance fields, excluding more detailed information or alternative data. Third, backtesting assumes simplified execution (the daily close-to-close with linear transaction costs). Future research can expand the framework to include additional assets, such as the S&P 500, to increase the number of examples and take advantage of the way different assets are connected. It can also adopt higher-quality data sources and test diverse market conditions to enhance the framework's robustness and generalizability

References

[1] Kumar, M., & Thenmozhi, M. (2014). Forecasting stock index returns using ARIMA-SVM, ARIMA-ANN, and ARIMA-random forest hybrid models. International Journal of Banking, Accounting and Finance. https: //www.inderscienceonline.com/doi/abs/10.1504/IJBAAF.2014.064307

[2] Bucci, A. (2020). Realized Volatility Forecasting with Neural Networks. Journal of Financial Econometrics. https: //academic.oup.com/jfec/article/18/3/502/5856840

[3] Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. Proceedings of the AAAI Conference on Artificial Intelligence, 35(12), 11106-11115. https: //doi.org/10.1609/aaai.v35i12.17325

[4] Hu, X.-P. (2021). Stock Price Prediction Based on Temporal Fusion Transformer. 2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI). https: //ieeexplore.ieee.org/document/9731073/

[5] Chantrasmee, C., Jaiyen, S., Chaikhan, S., & Wattanakitrungroj, N. (2024). Stock Trading Signal Prediction Using Transformer Model and Multiple Indicators. 2024 28th International Computer Science and Engineering Conference (ICSEC). https: //ieeexplore.ieee.org/document/10770734/

[6] Engelberg, J. E., McLean, R., Pontiff, J., & Ringgenberg, M. C. (2021). Do Cross-Sectional Predictors Contain Systematic Information? Journal of Financial and Quantitative Analysis. https: //www.cambridge.org/core/journals/journal-of-financial-and-quantitative-analysis/article/do-crosssectional-predictors-contain-systematic-information/C842FFBA1F84B696CDAFCE4F615C7339

[7] Aksehir, Z. D., & Kiliç, E. (2024). Analyzing the critical steps in deep learning-based stock forecasting: a literature review. PeerJ Computer Science. https: //peerj.com/articles/cs-2312/