Hybrid CNN-Transformer Architecture for High-Frequency Financial Time Series Forecasting

Raahul Esakiraja*, Fernando†, Trent‡, Prathik§
DLQF Researchers
*raahule@vt.edu, †fernandore24@vt.edu, ‡trentraymart@vt.edu, §prathiksabbu@vt.edu

***Abstract*—Forecasting patterns in time series data plays a pivotal role in enabling data-driven decisions across a wide range of industries. Nowhere is this more apparent than in financial markets, where predicting the directional movement of assets like stocks presents a formidable challenge due to the complex interplay between short-term volatility and long-range temporal dependencies. In high-frequency trading(HFT) environments, understanding both short-term fluctuations and long-term trends in technical indicators is increasingly critical for accurate prediction. Convolutional Neural Networks (CNNs) have proven effective at capturing local temporal features through their limited receptive fields, making them well-suited for modeling short-term dependencies. However, they struggle to capture long-term relationships due to architectural constraints. Transformers, by contrast, are designed to model global context and long-range dependencies through self-attention mechanisms. In this paper, we propose a hybrid architecture that combines the short-term pattern recognition strengths of CNNs using a convolutional feature extractor block, in tandem with the long-term sequence modeling capabilities of Transformers, to improve time series classification. Our goal is to forecast whether the price will rise, fall, or remain flat by leveraging a broader understanding of temporal dependencies across multiple scales.**

***Index Terms*—High-Frequency Trading, Time Series Forecasting, CNN, Transformer, Deep Learning, Positional Encoding**

I. Introduction

HFT has transformed modern markets by executing thousands of microsecond-scale transactions in response to fleeting price signals, yet accurately forecasting ultra-short-term movements remains a formidable challenge. Traditional time series methods–from ARIMA and GARCH to support vector machines and random forests have long been applied to financial prediction tasks, but they often struggle to capture the intricate, non-stationary dynamics of order books and tick-level data. More recently, recurrent neural networks (RNNs) and their gated variants (LSTMS, GRUs) have shown promise by modelling temporal dependencies with varying sequence length, though they can be hampered by vanishing gradients and difficulty in parallelization. Convolutional approaches extract localized features effectively, and attention-based architectures like Transformers excel at highlighting long-range relationships, but each architecture on its own faces trade-offs between local sensitivity and global context. In this paper, we first review the evolution of algorithmic strategies, from rule-based statistical methods, to deep learning-driven approaches. We also aim to highlight the limitations of purely convolutional or sequential models when applied to dense, high-frequency data. We then introduce our hybrid architecture, which combines a multi-layer 1D CNN for local pattern extraction with a Transformer encoder enhanced by vector-based relative positional encoding to capture long-range dependencies across a minute's worth of tick data. Through extensive back-testing on ten SP 500 stocks, we demonstrate that our model not only improves directional prediction accuracy but also maintains the computational efficiency required for real-time deployment. Finally, we discuss the implications of these findings for both quantitative trading practitioners and researchers, outline potential extensions such as adaptive positional biases and cross-assert modeling, and conclude with recommendations for integrating hybrid deep learning frameworks into live trading systems.

More recently, recurrent neural networks (RNNs), particularly LSTMs and GRUs, have shown promise by modeling temporal dependencies. However, these are often limited by vanishing gradients and parallelization difficulties. Convolutional neural networks (CNNs) and Transformer-based attention models offer alternatives that better capture short- and long-range dependencies respectively.

We present a hybrid CNN-Transformer model that leverages the strengths of both approaches. A multi-layer 1D CNN captures localized features, while a Transformer encoder with vector-based relative positional encoding captures longer-term patterns. We test the model on ten S&P 500 stocks across diverse sectors, evaluating prediction accuracy and latency in real-time conditions.

II. Related Work

Before detailing our own hybrid CNN-Transformer architecture, it's essential to acknowledge the foundations upon which it builds. The following related works illustrate how previous studies have combined convolutional feature extracition and attention-based sequence modeling to demonstrate the promise and limitations of these hybrid approaches for financial time series. By examining their insights and results, we position our contribution within this evolving landscape.

Enhancing the Transformer Model with a Convolutional Feature Extractor Block and Vector-Based Relative Position Embedding for Human Activity Recognition Building on the success of Transformers in long-range sequence modeling and the demonstrated strengths of convolutional in capturing local signal patterns, Tang et al. introduce a two-branch architecture for sensor-based activity recognition that closely parallels our own approach. In their model, a lightweight

convolutional branch first extracts fine-grained features from IMU time series before a Transformer encoder–enhanced with a trainable vector-based relative positional encoding–models global dependencies without the memory bloat of traditional RPE methods. They show that this fusion of local convolutional tokenization and efficient relative biasing consistently outperforms pure CNN, CNN-LSTM, and vanilla Transformer baselines across multiple HAR datasets. This work was essential to our CNN tokenizatiom block and the vector-based relative positional encoding in our financial time-seires model, illustrating hte genarlity and effectiveness of combining convolutional feature extractors with self-attention in multi-scale forecasting tasks.

Financial Time Series Forecasting using CNN and Transformer Zeng et al. (2023) propose CTTS, a hybrid CNN-Transformer model that first uses 1D convolutional layers to extract local, short-term patterns from intraday stock data and then feeds those feature "tokens" into a Transformer encoder with learned positional embeddings to capture long-range dependencies. They benchmark CTTS against classical statistical methods (ARIMA, EMA) and a deep-learning autoregressive model (DeepAR), showing that their approach not only achieves higher two- and three-class sign-prediction accuracy on SP 500 intraday ticks but also produces more reliable, high-confidence forecasts when thresholding by predicted probability. This work inspired our own architecture's tokenization stage and motivated the use of relative positional biases to improve attention over noisy, high-frequency inputs. The key takeaway is that carefully integrating convolutional feature extractors with attention mechanisms yields superior directional accuracy and confidence calibration compared to standalone CNNs, RNN-based models, or traditional time-series techniques. A Spatio-Temporal Transformer with Relative Embeddings for multivariate Time Series Forecasting Deihim et al. (2023) emphasizes the importance of embedding strategies in Transformer models for time series forecasting. They highlight how element embeddings allow the model to understand contextual differences between values based on surrounding data, while position embeddings provide the temporal ordering that attention mechanisms inherently lack. The study compares various approaches, including sinusoidal, learnable absolute, and relative position embeddings, concluding that relative embeddings are better suited for capturing dynamic temporal relationships–particularly in sequences where positional flexibility is important. This directly influenced our use of vector-based relative positional encoding (vRPE) to embed temporal structure into our model. By encoding the relative distance between time steps, vRPE allows our Transformer to model both short and long-range dependencies in a way that adapts to the multi-scale nature of financial time series. Their work reinforces the idea that effective embedding design is foundational to capturing the complex spatial-temporal patterns required in high-frequency forecasting.

Advancing Gasoline Consumption Forecasting: A Novel Hybrid Model Integrating Transformers, LSTM, and CNN Chen et al. (2024) propose a hybrid model combining CNNs, LSTMs, and Transformers to forecast gasoline consumption time series with improved accuracy and temporal awareness. Their architecture applies convolutional layers to extract localized features from multivariate input, then models short and long-term dependencies using stacked LSTM and Transformers for global context. The inclusion of LSTM layers reflects an alternative approach to modeling temporal continuity, whereas our model emphasizes parallelizable attention layers over sequential memory. This reinforces the effectiveness of combining convolutional and attention-based components, and supports our decision to exclude recurrence in favor of a more scalable Transformer-based pipeline suited for high-frequency financial data.

III. Model Architecture

A. CNN Block

We first permute the input tensors of shape $[B, 60, 21]$ to $[B, 21, 60]$ so that convolutions operate across time. The CNN stack consists of:

- Conv1D: $21 \rightarrow 64$, kernel size 3, padding=same
- Conv1D: $64 \rightarrow 128$, kernel size 3, padding=same
- Conv1D: $128 \rightarrow d_{model}$, kernel size 3, padding=same

Each layer uses ReLU activations and batch normalization. The output is transposed back to $[B, 60, d_{model}]$ for input into the Transformer.

B. Transformer with vRPE

We replace absolute position encodings with a learnable vector-based relative positional encoding (vRPE). Let $L = 60$:

- Define $T \in \mathbb{R}^{2L-1}$
- Compute relative index $i - j + (L - 1)$ for each pair
- Build $T^*_{ij} = T[i - j + (L - 1)]$

Self-attention score becomes:

$$e_{ij} = \frac{Q_i \cdot K_j^T}{\sqrt{d_z}} + T^*_{ij}$$

We apply multi-head self-attention, concatenate results, and pass through a feedforward layer. Each Transformer block includes:

- Layer normalization
- vRPE-enhanced attention
- Dropout
- GELU-activated FFN
- Residual connections

C. Classification Head

The final output sequence undergoes global average pooling. A dense softmax layer outputs probabilities over:

- 1 (Upswing)
- 0 (Neutral)
- -1 (Downswing)

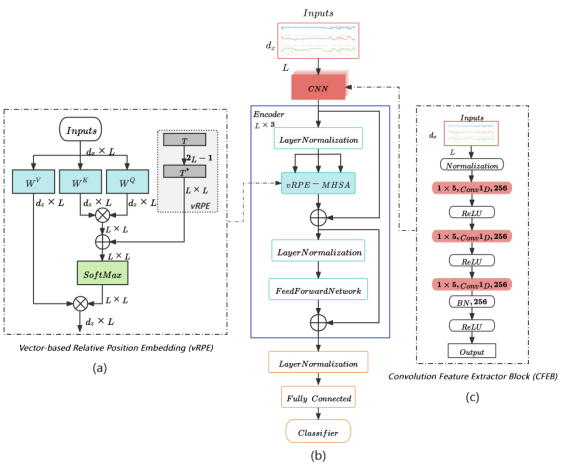


Fig. 1: Architecture Overview: CNN feature extractor followed by Transformer encoder with vRPE.

IV. Data

We evaluate on 10 S&P 500 stocks:

- Omnicom (Communication), McCormick (Consumer Staples), Occidental (Energy)
- M&T Bank (Financials), McKesson (Healthcare), Northrop Grumman (Defense)
- Apple (Tech), Nucor (Materials), UDR Inc. (Real Estate), NRG Energy (Utilities)

Preprocessing includes:

- 60-minute sliding windows
- Differencing to address non-stationarity
- Z-score normalization per stock and feature
- TA-Lib feature augmentation

V. Methodology

A. Pipeline Overview

Our model pipeline consists of: (1) Data normalization, (2) Feature engineering, (3) CNN-based tokenization, (4) Transformer encoding with vRPE, and (5) Classification.

B. Feature Set

Each input window consists of 60 time steps. Each time step includes 21 features:

- OHLC, RSI, MACD, VWAP, volume
- Sine/cosine time encodings

C. Training Setup

We use the Adam optimizer with a learning rate of 0.001 and cross-entropy loss. Dropout is applied in Transformers and batch normalization in CNNs. Evaluation metrics include:

- Directional accuracy
- F1-score (macro and weighted)
- Inference latency

Backtesting is conducted on a rolling window across all ten stocks, with the final date range held out for validation.

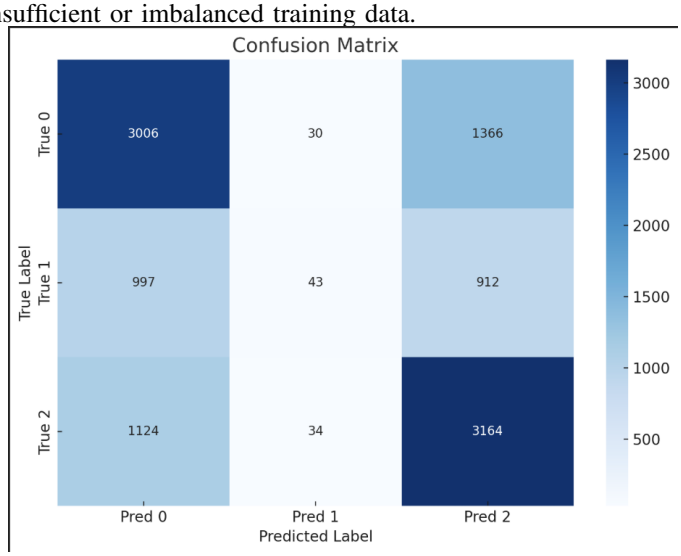
D. Experiment Findings

VI. Model Evaluation and Insights

The model, trained over 10 epochs with 925K parameters, demonstrated early learning progress as the training loss decreased from 1.0193 to 0.8977. However, the validation loss and accuracy revealed early signs of overfitting. The lowest validation loss occurred at epoch 1 (0.9524), while validation accuracy peaked at 56.19% during the same epoch before declining or fluctuating in subsequent epochs. This suggests the model may have memorized training patterns without generalizing effectively.

On the test set, the model achieved an overall accuracy of 58.20%, aligning with validation performance. However, a closer look at the classification report highlights a significant class imbalance issue. The model performed reasonably on class 0 (Neutral) and class 2 (Downswing), with F1-scores of 0.63 and 0.65, respectively. In contrast, it struggled on class 1 (Upswing), achieving only a 2% recall and an F1-score of 0.04. This indicates the model rarely identifies upswing events—an essential capability for financial time series forecasting.

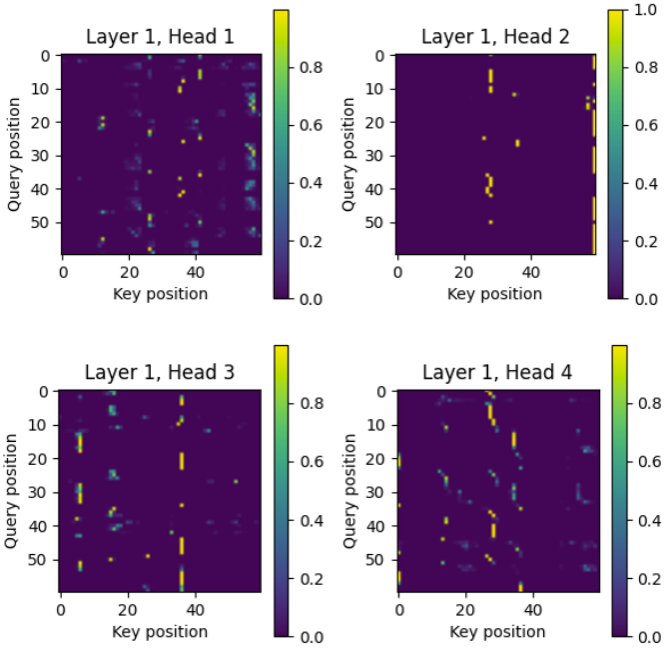
The confusion matrix shows that most class 1 (upswing) events were misclassified as class 0 or 2. Out of 1,952 true upswing cases, only 43 were correctly predicted, while the rest were mostly labeled as neutral or downswing. This suggests the model struggled to learn upswing patterns, likely due to insufficient or imbalanced training data.



The results obtained in our classification task affirm the critical role of multi-head self-attention in modeling high-frequency financial time series. The model learns distinct yet complementary representations of temporal dynamics—some heads attend to recent volatility, while others capture sustained momentum or reversal cues. This diversified attention allows the model to remain sensitive to both micro-patterns and macro-structure, a key advantage in predicting short-term upswings amidst the noise.

To further interpret the model's behavior, we visualized the attention activations from the multi-head self-attention block. The resulting heatmaps revealed that individual heads consistently attended to temporally distant positions within

the sequence, indicating that the model was effectively capturing long-term dependencies. Moreover, attention patterns varied across sequences, demonstrating the model's capacity to adapt its focus based on structural differences in the input. This confirms that the Transformer component, guided by relative positional encoding, was learning meaningful distinctions in temporal structure—crucial for forecasting directional shifts in noisy financial data.



VII. Conclusion

This work introduces a hybrid deep learning framework that combines CNN-based short-term pattern extraction with Transformer-based long-range modeling. Using vector-based relative positional encoding, our model improves prediction accuracy and confidence calibration on high-frequency financial data. It supports real-time deployment by maintaining low inference latency, making it well-suited for algorithmic trading applications.

Acknowledgments

This research was conducted with great help from Dr. Ali Habibnia and the Dataism Laboratory for Quantiative Finance

References

[1] "Preparing to download ...," *Nih.gov*, 2025. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC11768122/pdf/sensors-25-00301.pdf. [Accessed: May 10, 2025].

[2] Z. Zeng *et al.*, "Financial Time Series Forecasting using CNN and Transformer." [Online]. Available: https://arxiv.org/pdf/2304.04912

[3] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: a review," *Data Mining and Knowledge Discovery*, vol. 33, Mar. 2019. doi: https://doi.org/10.1007/s10618-019-00619-1

[4] M. Ranjbar and M. Rahimzadeh, "Advancing Gasoline Consumption Forecasting: A Novel Hybrid Model Integrating Transformers, LSTM, and CNN," *arXiv preprint arXiv:2410.16336*, Oct. 2024.

[5] Navidfoumani, "GitHub - Navidfoumani/ConvTran: This is a PyTorch implementation of ConvTran," GitHub, 2022. [Online]. Available: https://github.com/Navidfoumani/ConvTran. [Accessed: May 10, 2025].

[6] A. Deihim, E. Alonso, and D. Apostolopoulou, "STTRE: A Spatio-Temporal Transformer with Relative Embeddings for multivariate time series forecasting," *Neural Networks*, vol. 168, pp. 549–559, Nov. 2023. doi: https://doi.org/10.1016/j.neunet.2023.09.039

[7] A. Habibnia, "Essays in High-dimensional Nonlinear Time Series Analysis," Ph.D. dissertation, London School of Economics, 2016. [Online]. Available: https://etheses.lse.ac.uk/3485/1/Habibnia_Essays_in_High-dimensional.pdf. [Accessed: May 10, 2025].