TLOB: A Novel Transformer Model with Dual Attention for Stock Price Trend Prediction with Limit Order Book Data

Leonardo Berti
leonardo.berti@tum.de
Technical University of Munich, Germany

Gjergji Kasneci
gjergji.kasneci@tum.de
Technical University of Munich, Germany

Abstract

Stock Price Trend Prediction (SPTP) based on Limit Order Book (LOB) data is a fundamental challenge in financial markets. Despite advances in deep learning, existing models fail to generalize across different market conditions and struggle to reliably predict short-term trends. Surprisingly, by adapting a simple MLP-based architecture to LOB, we show that we surpass SoTA performance; thus, challenging the necessity of complex architectures. Unlike past work that shows robustness issues, we propose TLOB, a transformer-based model that uses a dual attention mechanism to capture spatial and temporal dependencies in LOB data. This allows it to adaptively focus on the market microstructure, making it particularly effective for longer-horizon predictions and volatile market conditions. We also introduce a new labeling method that improves on previous ones, removing the horizon bias. We evaluate TLOB's effectiveness using the established FI-2010 benchmark, which exceeds the state-of-the-art by an average of 3.7 F1-score(%). Additionally, TLOB shows improvements on Tesla and Intel with a 1.3 and 7.7 increase in F1-score(%), respectively. Additionally, we empirically show how stock price predictability has declined over time (-6.68 absolute points in F1-score(%)), highlighting the growing market efficiencies. Predictability must be considered in relation to transaction costs, so we experimented with defining trends using an average spread, reflecting the primary transaction cost. The resulting performance deterioration underscores the complexity of translating trend classification into profitable trading strategies. We argue that our work provides new insights into the evolving landscape of stock price trend prediction and sets a strong foundation for future advancements in financial AI. We release the code at https://github.com/LeonardoBerti00/TLOB.

CCS Concepts

• **Computing methodologies** → **Neural networks**.

Keywords

Stock Price Prediction, Limit Order Book, Transformer

1 Introduction

Over the past few decades, the global financial landscape has undergone a profound transformation, transitioning from manual trading operations to sophisticated electronic platforms. This evolution has been so significant that by 2020, electronic trading accounted for over 99% of equity shares traded in the United States, a stark contrast to just 15% in 2000 [25]. At the heart of this revolution lies the electronic Limit Order Book (LOB), a dynamic data structure that has become the cornerstone of modern financial markets. In today's competitive financial world, the majority of the markets utilize electronic LOBs to record trades. The continuous inflow of limit orders, organized by price levels, creates a dynamic structure that evolves over time, reflecting the real-time balance of supply and demand. However, this multidimensional structure, which spans price levels and volumes, presents complex challenges for understanding market behavior, forecasting stock price trends, and simulating realistic market conditions. The non-stationary nature of LOB data, characterized by its stochastic behavior, makes modeling stock price movements challenging. Traditional statistical methods fail to capture these complexities, especially when attempting to predict short-term price trends. However, recent advancements in deep learning have opened new avenues for tackling these challenges, offering the ability to model the non-linear relationships and temporal dependencies inherent in LOB data.

Stock Price Trend Prediction (SPTP)[1] remains one of the most challenging and economically significant problems in financial markets, attracting significant attention from academic researchers and industry practitioners. One prominent application of SPTP, particularly utilizing Limit Order Book (LOB) data, lies within high-frequency trading, where algorithms attempt to capitalize on short-term price movements. Predicting future market movements is a highly challenging task due to the complexity, non-stationarity, and volatility of financial markets. However, with the growing availability of Limit Order Book (LOB) data and advancements in deep learning, new opportunities have emerged to improve the accuracy of these predictions. This paper explores the application of deep learning models to SPTP using Limit Order Book (LOB) data, which provides the most granular and complete information on stock trades. Financial markets do not exist in a vacuum; they are continuously shaped by the actions and expectations of countless participants who, according to the Efficient Market Hypothesis (EMH), collectively incorporate all available information into asset prices. When models discover a predictive pattern and traders act

---

[1]In the literature it is also referred to as mid-price movement prediction.

on it, the anomaly is quickly competed away, causing a paradox: successful signals sow the seeds of their own demise. Over time, greater liquidity, advanced trading technologies, and the proliferation of algorithmic strategies intensify this effect, i.e., any exploitable signal becomes visible in execution data and erodes more rapidly. Consequently, the apparent decline in forecast accuracy in our findings aligns with EMH principles: as soon as new patterns are detected and exploited, the relentless engine of arbitrage drives markets back toward efficiency. This interplay underlines why forecasting often becomes more difficult the farther we move from idealized, less liquid markets (like FI-2010) toward active, high-efficiency markets (like NASDAQ), thereby illustrating a core tension between the pursuit of alpha and the self-correcting nature of competitive markets. Traditional approaches relied on technical analysis and statistical methods, but recent years have seen a shift toward more sophisticated deep learning methods. A lot of different types of deep learning architectures have been utilized to tackle the SPTP tasks. Tsantekidis et al. utilized Long Short-Term Memory (LSTM) layers [41] and Convolutional Neural Networks (CNNs) [42, 43]. Zhang et al. [48] introduced the DeepLOB model, which leverages LOB data to predict mid-price movements using a combination of convolutional and LSTM layers. Recent work [33] has highlighted the limitations of existing models, particularly their lack of robustness and generalizability when applied to diverse market conditions and more efficient stocks. In this paper, we address these limitations by proposing TLOB, a transformer-based approach, that outperforms all the existing models on both benchmark and real-world datasets, paving the way for more reliable SPTP applications. We introduce also an MLP-based model to show that a simple architecture, based on fully connected layers and GeLU activation function, can outperform all the SoTA models. We list our contributions:

(1) **Novel Architecture Proposals**: We introduce two new deep learning models:
   - **MLPLOB**: A simple yet effective MLP-based model inspired by recent advances in the deep learning literature.
   - **TLOB**: A transformer-based approach that leverages dual attention mechanisms for both temporal and spatial relationships in LOB data.
(2) **Comprehensive Evaluation**: We conduct extensive experiments on both the benchmark FI-2010 dataset and a real-world NASDAQ dataset composed of Tesla and Intel stocks, with several baselines, providing insights into model performance across different market conditions and time horizons. We also perform an ablation study investigating the design choices of TLOB.
(3) **New Labeling Methods**: We introduce a new labeling method that improves on previous ones, removing the horizon bias.
(4) **Historical Comparison**: We examine whether stock price prediction has become more difficult over time by comparing model performance on historical data from different periods.
(5) **Alternative Threshold Definition**: We propose and evaluate a novel approach to defining trend classification thresholds based on average spread, directly incorporating the primary transaction cost into the prediction framework.

2 Background

In the contemporary, highly competitive financial landscape, the predominant mechanism for recording and managing market transactions is the electronic Limit Order Book (LOB). Within a limit order book market, traders can submit orders to buy or sell a specified quantity of an asset at a predetermined price. Three primary order types are prevalent in such markets: (1) **Market orders**, which are executed immediately at the best available price with a predetermined quantity; (2) **Limit orders**, allows traders to decide the maximum (in the case of a buy) or the minimum (in the case of a sell) price at which they want to complete the transaction. A quantity is always associated with the specified price; and (3) **Cancel orders** (alternatively referred to as deletions), which serve to remove an active limit order.

The LOB is a data structure that maintains and matches active limit orders and market orders in accordance with a predefined set of rules. This structure is transparently accessible to all market participants and is subject to continuous updates with each event, including order placement, modification, cancellation, and execution. The most widely adopted mechanism for order matching is the Continuous Double Auction (CDA) [3]. Under the CDA framework, orders are executed whenever the best bid (the highest price a buyer is willing to offer) and the best ask (the lowest price a seller is willing to accept) overlap. This mechanism facilitates continuous and competitive trading among market participants. The price of a security is commonly defined as the mid-price, calculated as the average of the best ask and best bid prices, with the difference between these prices representing the bid-ask spread.

Given that limit orders are organized into distinct depth levels, each comprising bid price, bid size, ask price, and ask size, based on their respective prices, the temporal evolution of a LOB constitutes a complex, multidimensional temporal problem. Research on LOB data can be broadly categorized into four primary types: empirical analyses of LOB dynamics [5, 11], price and volatility forecasting [37, 48], stochastic modeling of LOB dynamics [12, 16], and LOB market simulation [7, 10, 26].

3 Related Work

The challenge of modeling the complex data structures and vast quantities associated with LOBs has spurred the development of deep learning algorithms for related modeling and forecasting tasks. In this section, we will summarize the State-of-The-Art (SoTA) deep learning models in the Stock Price Trend Prediction (SPTP) task, which consists of forecasting the direction of mid-price movements at a high-frequency resolution. Tsantekis et al.[42] (2017) utilize a Recurrent Neural Network (RNN) based on Long-Short Term Memory (LSTM) layers to predict mid-price movements. In the same year, the authors presented another approach [41], introducing a CNN-based model (CNN). Subsequently, the same group proposed two additional architectures in [43] (2020). The first focuses on capturing temporal dynamics from LOB data and correlating temporally distant features using convolutional layers. The second architecture, CNNLSTM, merges the CNN with an LSTM. The CNN initially extracts features from the LOB time series, which are then passed to the LSTM for classification. Tran et al.[39] (2018) proposed the Temporal Attention-Augmented Bilinear Layer (TABL)

for multivariate time series prediction. This architecture, applies a bilinear transformation to the input, capturing dependencies between features and over time. The authors extended this model in [40], introducing BINCTABL, which integrates a bilinear normalization layer into the architecture to address non-stationarity and magnitude disparity (for instance prices and sizes) within the time series data. Passalis et al.[32] (2019) introduced DAIN (Deep Adaptive Input Normalization), a three-step layer that adaptively normalizes data, instead of relying on fixed precomputed statistics. DAIN comprises three layers: a shifting layer, a scaling layer, and a gating layer, which suppresses irrelevant features by applying a sigmoid function. DAIN was integrated into various architectures, including MLPs [30], CNNs [41], and RNNs [9]. Zhang et al. [48] (2019) introduced DEEPLOB, which consists of three main blocks: convolutional layers, an Inception Module, and an LSTM layer. The convolutional layers and Inception Module extract relevant features, while the LSTM captures temporal dependencies. Two years later, Zhang et al.[47] (2021) extended DEEPLOB by adopting the attention [27] mechanism, creating DEEPLOBATT for multi-horizon forecasting. In this architecture, an encoder extracts features from LOB data, and an attention mechanism assigns weights to the hidden states, improving the processing of long input sequences. Finally, Kiesel et al.[24] (2022) introduced Axial-LOB, which uses axial attention [18] to factorize 2D attention into two 1D attention modules, one for the feature axis and one for the time axis. Prata et al. [33] evaluated 15 deep learning models for stock prediction using limit order book data, including all the models described above. Some performed well on the FI-2010 dataset, but most of them showed non-reproducible results. When trained and tested on a new dataset composed of NASDAQ stocks most failed to generalize, in particular the performances of every model flattened around 60 in F1-Score. BiN-CTABL ([40]) was the top performer, and attention-based models generally excelled. The main reason given by the authors is that NASDAQ stocks are more complex to forecast than Finnish ones and the SoTA models cannot capture this complexity. Furthermore, all models showed high sensitivity to hyperparameters and context. These results make the models unreliable for real-world use.

Another stream of research has focused on meta-learning Transformer models, i.e., TabPFN models [19, 20], that excel at "one forward-pass" inference on small tabular datasets, leveraging massive synthetic corpora of low-dimensional tasks to learn prior feature distributions. However, they become computationally prohibitive for large-scale LOB data, where millions of high-dimensional observations strain both memory and processing. Consequently, TabPFN-based methods remain impractical for demanding real-world predictive tasks, such as SPTP, underscoring the need for more scalable deep learning architectures (e.g., LSTM, CNN, or specialized Transformers) tailored to extensive time series.

4 Task Definition

We represent the evolution of a LOB as a time series $\mathbb{L}$, where each $\mathbb{L}(t) \in \mathbb{R}^{4L}$ is called a LOB record, for $t = 1, \ldots, N$, with N being the number of LOB observations and L the number of levels. In particular,

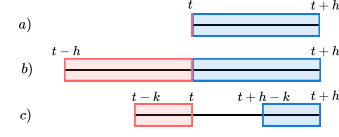$$\mathbb{L}(t) = \left(P^{ask}(t), V^{ask}(t), P^{bid}(t), V^{bid}(t)\right), \tag{1}$$

Figure 1: Comparison of three labeling methods. t is the current timestamp, k is the smoothing window length, and h is the prediction horizon. In our proposed method (c), k and h are defined independently, providing a more flexible and unbiased approach.

where $P^{ask}(t)$ and $P^{bid}(t) \in \mathbb{R}^L$ are the prices at levels 1 through L, and $V^{ask}(t)$ and $V^{bid}(t) \in \mathbb{R}^L$ are the corresponding volumes. **Trend Definition** We employ a ternary classification system for price trends: U ("upward") denotes an increasing price trend, D ("downward") indicates a decreasing trend, and S ("stable") represents price movements with only minor variations.

In equity markets, mid-prices are generally considered the most reliable single-value indicator of actual stock prices. However, owing to inherent market fluctuations and exogenous shocks, mid-prices can exhibit considerable volatility. Consequently, labeling consecutive mid-prices (p_t, p_{t+1}) often results in noisy labels.

To mitigate this, many labeling strategies employ smoother mid-price functions, averaging prices over a chosen "window length" to reduce short-term noise and better reflect persistent directional moves. An example of this approach appears in [31], detailed in Section 6.2.

However, as shown by Zhang et al. [48] (Fig. 2), smoothing only the future prices can lead to instability in trading signals. This instability often causes redundant trading actions and higher transaction costs. To address this, Tsantekidis et al. [41] proposed also smoothing past prices. They define:

$$l(t, k) = \frac{m_+(t, k) - m_-(t, k)}{m_-(t, k)} \quad \text{where} \tag{2}$$

$$m_+(t, k) = \frac{1}{k+1} \sum_{i=0}^{k} p(t + i) \quad \text{and} \tag{3}$$

$$m_-(t, k) = \frac{1}{k+1} \sum_{i=0}^{k} p(t - i), \tag{4}$$

noting that i runs from 0 to k, so there are $(k + 1)$ terms in the sum. A key drawback is that the window length k coincides with the prediction horizon h. This can bias the labels: for instance, a horizon of $h = 2$ may not provide enough smoothing, whereas a large horizon might over-smooth price moves.

To overcome this, we propose a more general labeling strategy that dissociates k from h. Specifically, we define:

$$w_+(t, h, k) = \frac{1}{k+1} \sum_{i=0}^{k} p(t + h - i) \tag{5}$$

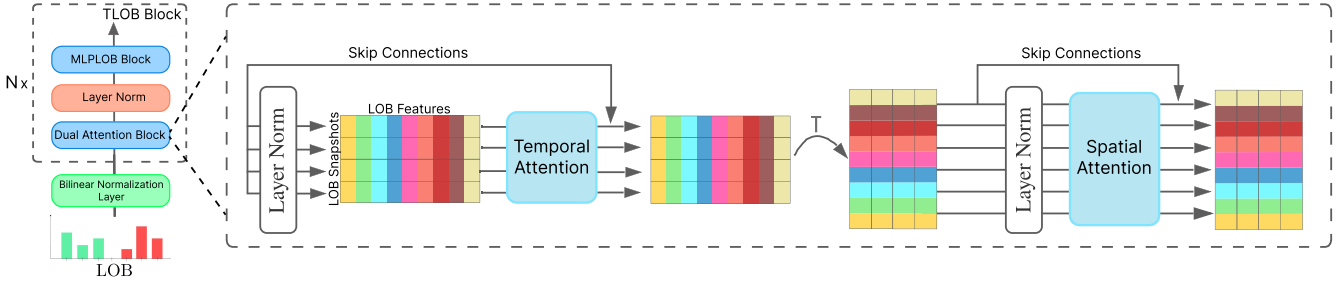$$w_-(t, h, k) = \frac{1}{k+1} \sum_{i=0}^{k} p(t - i). \tag{6}$$

Figure 2: TLOB architecture overview. The model leverages Temporal Self-Attention and Feature Self-Attention within each TLOB block to capture time-wise and spatial relationships in Limit Order Book data. Each block is preceded by Bilinear Normalization to address non-stationarity, followed by an MLPLOB block.

The percentage change is then

$$l(t, h, k) = \frac{w_+(t, h, k) - w_-(t, h, k)}{w_-(t, h, k)}. \quad (7)$$

We classify a trend as *upward* if $l(t, h, k) > \theta$, *downward* if $l(t, h, k) < -\theta$, and *stable* if $-\theta \leq l(t, h, k) \leq \theta$. The threshold θ is often chosen to balance the three classes rather than to reflect trading costs. We argue, however, that relating θ to transaction costs can better align trend predictions with profitability. Thus, in Section 7.4, we examine setting θ to the average spread (the difference between the best bid and ask prices) as a percentage of the mid-price[2], since the spread represents the main transaction cost.

Figure 1 illustrates all three approaches. For a fair comparison with existing literature, we adopt the original labeling method in our FI-2010 experiments and use our new labeling strategy for Intel and Tesla data, where the more general approach better handles varying horizons.

5 Models

We propose two novel deep learning models for Stock Price Trend Prediction (SPTP) using Limit Order Book (LOB) data. The first, called **MLPLOB**, is a simple MLP-based model. The second, **TLOB**, leverages a dual-attention Transformer-based approach. Both models take as input a sequence of LOB time series consisting of the last T LOB snapshots for 10 LOB levels.

5.1 MLPLOB

A key finding from the benchmark study by Prata et al. [33] reveals that, despite the proliferation of specialized deep learning architectures for SPTP, their performance often converges toward low values when tested on diverse and complex datasets. Inspired by the work of Tolstikhin et al. [38] and Zeng et al. [46], who demonstrated that simple MLP-based models can perform as well as state-of-the-art (SoTA) methods in certain domains, we develop an MLP-based architecture for SPTP with LOB data, called *MLPLOB*.

Architecture Overview. MLPLOB is composed of multiple blocks, each containing two types of MLP layers:

(1) *Feature-Mixing MLPs*, which operate along the feature axis.
(2) *Temporal-Mixing MLPs*, which operate along the time axis.

This design aims to capture both spatial and temporal relationships in LOB data–characteristics that Sirignano and Cont [36, 37] identified as fundamental to LOB dynamics and modeling.

Each MLP layer consists of two fully connected layers, mirroring the MLP component used in Transformer architectures [44]. Initially, the input sequence is projected linearly into a tensor $\mathbf{X} \in \mathbb{R}^{T \times N}$, where N is a chosen hyperparameter.

Feature-Mixing MLPs. We apply a feature-mixing MLP row by row (*i.e.*, for each time step i). Formally,

$$\mathbf{U}_{i,*} = \sigma\Big(\text{LayerNorm}\big(\sigma(\mathbf{X}_{i,*} \mathbf{W}_1) \mathbf{W}_2 + \mathbf{X}_{i,*}\big)\Big) \quad \text{for } i = 1, \ldots, T, \quad (8)$$

where σ is the GeLU activation function [17], and LayerNorm denotes layer normalization.

Temporal-Mixing MLPs. Next, we transpose the resulting tensor $\mathbf{U}$ and apply a temporal-mixing MLP column by column (*i.e.*, for each feature dimension j):

$$\mathbf{Z}_{*,j} = \sigma\Big(\text{LayerNorm}\big(\sigma(\mathbf{U}_{*,j} \mathbf{W}_3) \mathbf{W}_4 + \mathbf{U}_{*,j}\big)\Big) \quad \text{for } j = 1, \ldots, N. \quad (9)$$

Model Simplicity and Isotropic Design. The MLPLOB architecture relies only on matrix multiplications, reshaping operations, and scalar nonlinearities. It also adopts an *isotropic design*, wherein each block (beyond the initial projection) has a constant dimensionality. This contrasts with the pyramidal layouts found in many CNNs (which reduce spatial resolution while increasing channel depth). Notably, isotropic designs are also common in Transformers and Recurrent Neural Networks (RNNs).

Final Prediction. After several blocks of feature and temporal mixing, MLPLOB performs dimensionality reduction to collapse all features into a single vector, which then passes through several fully connected layers that gradually diminish the vector dimension and a final standard classification head. The network outputs the directional trend (up, down, or stable) for the final time step. Our primary objective in devising MLPLOB is to show that a carefully structured MLP-based model can match or exceed more complex architectures in the SPTP task. The same method is also applied to TLOB.

5.2 TLOB

The Transformer architecture [44] has led to major breakthroughs in deep learning, notably in natural language processing [6, 23]

---

[2]Expressing the spread as a percentage of the mid-price preserves consistency with $l(t, h, k)$, which is also a percentage.

and time-series modeling [45]. A key advantage is the ability to capture long-range dependencies without suffering as much from vanishing gradients or forgetting, and performance typically scales favorably with increased data [22]. Because massive volumes of financial data (in the terabyte range) are available, Transformers are well-positioned for LOB modeling.

Dual-Attention Blocks. We propose *TLOB*, a Transformer-based architecture specifically designed for Limit Order Book data. Each TLOB block contains:

(1) *Self-Attention over LOB Snapshots (Temporal Axis)*, computes attention values between different LOB snapshots, capturing time-wise dependencies among consecutive snapshots.

(2) *Self-Attention over LOB Features (Spatial Axis)*, computes attention values between LOB features, capturing spatial relationships among different price-volume features.

(3) An *MLPLOB block*, which replaces the usual Transformer feed-forward network to enhance the model's capacity for combining spatial and temporal signals.

The architecture is shown in Fig. 2.

Temporal vs. Feature Attention. While standard Transformers [44] process tokens along a single dimension, LOB data naturally requires both temporal and spatial dependencies to be learned [36, 37]. For instance, time-step t can reveal how deeper or shallower levels relate to one another, as well as how trends evolve over past snapshots. Hence, *dual-attention* explicitly addresses these two axes of variation. To investigate the importance of each type of attention layers we performed an ablation study (Section 7.5).

Bilinear Normalization Layer. To address non-stationarity and magnitude disparity (prices and sizes) in financial time series, we employ a Bilinear Normalization layer [40] as the initial layer. Unlike conventional z-score normalization, which can fail under distribution shifts at inference time, bilinear normalization adapts to batch-specific statistics, maintaining robust performance even when market conditions change. The same layer is also used in MLPLOB.

Positional Encoding. Because self-attention is permutation-invariant, we incorporate sinusoidal positional embeddings [44] to preserve the chronological structure within each LOB window. This embedding ensures that TLOB respects the temporal ordering of snapshots, which is crucial for modeling price evolution.

By blending two distinct self-attention operations (temporal first, then spatial) with an MLPLOB feed-forward component, TLOB is designed to capture the complex market microstructure present in LOB data. Its Transformer foundation enables effective scaling for large datasets, while the dual-attention mechanism better handles the fine-grained feature interactions and sequence dependencies characteristic of financial time series.

6 Experiments

We conduct a comprehensive evaluation of MLPLOB and TLOB model training and testing on both the Benchmark FI-2010 dataset and the TSLA-INTC dataset, composed of Tesla and Intel. TLOB and MLPLOB surpass SoTA performances on every dataset and every horizon. TLOB performs the best on larger horizons, while MLPLOB performs the best on the shorter ones. Our experiments extend beyond merely demonstrating the state-of-the-art performance of

TLOB, aiming to address several critical research questions: (1) Are stock prices harder to forecast than in the past? (2) What if we choose θ equal to the average spread? (3) Are temporal and spatial attention necessary? Through these investigations, we seek not only to validate our models' predictive capabilities but also to contribute to the broader understanding of deep learning applications in financial forecasting.

6.1 TSLA-INTC Dataset

In the majority of state-of-the-art (SoTA) research within the domain of Deep Learning applied to LOB data, researchers typically employ one, two, or three stocks [10, 21, 26, 29, 34, 35], predominantly from the technology sector. Adhering to this established practice, we construct a LOB dataset comprising two NASDAQ-listed stocks namely, Tesla and Intel – spanning the period from January 2nd to January 30th, 2015. We posit that stylized facts and market microstructure characteristics exhibit independence from individual stock behaviors (as demonstrated in [4, 5, 13, 16][3]), thereby rendering specific stock attributes non-critical to the analysis. The dataset encompasses 20 order book files for each stock, corresponding to each trading day, resulting in a total of approximately 24 million samples. Each order book sample is represented as a tuple $\left(P^{ask}(t), V^{ask}(t), P^{bid}(t), V^{bid}(t) \right)$, where $P^{ask}(t)$ and $P^{bid}(t) \in \mathbb{R}^L$ denote the prices at levels 1 through L, and $V^{ask}(t)$ and $V^{bid}(t) \in \mathbb{R}^L$ represent the corresponding volumes. The dataset is partitioned such that the initial 17 days are allocated for training, the 18th day for validation, and the final two days for testing. In Table 1 we present the main features of Tesla and Intel for January 2015. In Figure 3 we show the mid-price traces. As shown both the features and the mid-price traces are very different, making the evaluation more general. Unfortunately, we cannot make the dataset public for copyright reasons.

Sampling. Limit Order Book data, especially for liquid stocks, is massive, every day, hundreds of thousands of orders are placed for each stock. Furthermore, financial data are known to have a low signal-to-noise ratio [28]. Accordingly, it is unnecessary to consider every LOB update, so defining a valid sampling technique is essential. While time-based and event-based sampling methods [4] are used, they fail to capture the varying impact of transactions. In fact, single transactions can have very different impacts on the market. Volume-based sampling offers a solution by sampling the LOB after a predetermined volume of shares has been traded, thus reflecting the magnitude of market activity. Therefore, we adopted a sampling strategy based on trading volume, where snapshots of the Limit Order Book (LOB) are taken every 500 stocks traded. This method achieves a compromise between maintaining adequate temporal consistency within windows and ensuring significant variation between samples.

6.2 Benchmark dataset FI-2010

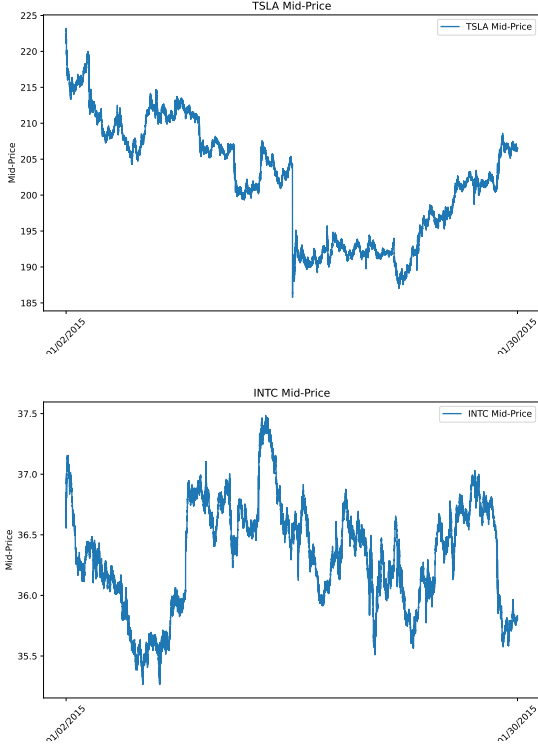Our model will be evaluated against SoTA models utilizing the FI-2010 benchmark dataset [31]. The FI-2010 dataset [31] is the most widely adopted LOB dataset within the field of deep learning

---

[3]These seminal works in finance elucidate the universal statistical properties of LOBs, transcending specific stocks and markets.
[4]FI-2010 uses this type of sampling, a LOB snapshot is taken every 10 events.

Table 1: Intel and Tesla main features for January 2015. Average liquidity is computed as the average quantity available in the first 10 LOB levels.

Stock	Daily Return (%)	Daily Volume	Avg. Spread	Avg. Liquidity
TSLA	-0.42 ± 2.84	$23,927,602 \pm 4,554,884$	0.16	$3,320$
INTC	-0.44 ± 1.66	$304,325,400 \pm 69,340,430$	0.01	$124,960$

Figure 3: Tesla and Intel mid-price traces for January 2015.

applications to limit order books [41, 42, 48, 49], particularly for forecasting endeavors. It comprises LOB data from five Finnish companies listed on the NASDAQ Nordic stock exchange: Kesko Oyj, Outokumpu Oyj, Sampo, Rautaruukki, and Wärtsilä Oyj. The data span ten trading days, from June 1st to June 14th, 2010, encompassing approximately 4 million limit order snapshots across ten levels of the LOB. The authors sampled LOB observations at intervals of ten events, resulting in a total of 394,337 samples. The label associated with each data point, indicative of mid-price movement, is determined by the percentage change between the prevailing mid-price:

$$p(t) = \frac{P^{ask}(t) + P^{bid}(t)}{2} \tag{10}$$

and the average of the subsequent h (chosen horizon) mid-prices:

$$m_+(t,k) = \frac{1}{k}\sum_{i=1}^{k} p(t+i) \tag{11}$$

The percentage change is thus defined as:

$$l(t) = \frac{m_+(t,k) - p(t)}{p(t)} \tag{12}$$

where k represents the window length, which in this instance also corresponds to the prediction horizon h. Labels are assigned as explained in 4. The dataset furnishes time series and corresponding class labels for five distinct horizons: $h \in H = \{10, 20, 30, 50, 100\}$. The dataset's authors employed a uniform threshold $\theta = 2 \times 10^{-3}$ across all horizons. The value is chosen to balance the classes for $h = 50$.

6.3 Experimental settings

For each dataset, we trained and tested the performance of each model on different horizons, namely 10, 20, 50, 100. All the experiments were carried out using an RTX 3090. Since the FI-2010 dataset also contains 104 handcrafted features derived from the LOB, we used them in both our models. This choice improved the performance of the F1-Score (%) by approximately 1. For Tesla and Intel, given the availability of message files containing the order information, we augmented the LOB snapshots by concatenating them with the corresponding orders. This integration was undertaken to incorporate additional information not present in the LOB. Consequently, this approach resulted in an approximate improvement of 1.5 in the F1-score (%) . We report the details on the hyperparameters search in the Appendix (A).

Baselines As comparative baselines, we employed 3 machine learning models: Support Vector Machine (SVM), Random Forest and XGBoost, and 8 deep learning SoTA LOB-based models: MLP, LSTM [41], CNN [42], CTABL [39], DAIN [32], CNNLSTM [43], DeepLOB [48] and BiN-CTABL [40]. Due to computational constraints, we selected the top two performing models from FI-2010, specifically DeepLOB, and BiNCTABL, and exclusively trained and tested these models with the TSLA-INTC dataset.

Trend Classification Threshold We remark that θ is the parameter that determines if a percentage change l_t is classified as an up, stable, or downtrend. For the TSLA-INTC dataset, to ensure balanced class distribution, we set θ equal to the mean percentage change. In Sec. 7.4 we explore an alternative approach to defining θ based on financial parameters rather than class balance optimization. For the FI-2010 dataset, we retained the original labels to maintain consistency with existing benchmark studies and previous works.

Metric We selected the F1-score as our primary performance metric because it captures both precision and recall in a single value. Accuracy is not a valid metric for our experiments because the classes are not balanced for each horizon. The F1-Score is robust to the class imbalance problem, which detrimentally affects the

accuracy. Finally, the F1-score is the most used metric in the SoTA papers tackling the SPTP task. For a comprehensive evaluation, we provide precision and recall curves in the Appendix (B).

7 Results

Table 2: F1-score on the FI-2010 dataset on four horizons. Bold values show the best scores.

Model	FI-2010 F1-Score (%) ↑			
	h = 10	h = 20	h = 50	h = 100
SVM	35.9	43.2	49.4	51.2
Random Forest	48.7	46.3	51.2	53.9
XGBoost	62.4	59.6	65.3	67.6
MLP	48.2	44.0	49.0	51.6
LSTM [42]	66.5	58.8	66.9	59.4
CNN [41]	49.3	46.1	65.8	67.2
CTABL [39]	69.5	62.4	71.6	73.9
DAIN-MLP [32]	53.9	46.7	61.2	62.8
CNNLSTM [43]	63.5	49.1	69.2	71.0
DeepLOB [48]	71.1	62.4	75.4	77.6
BiNCTABL [40]	81.1	71.5	87.7	92.1
MLPLOB	**81.64**	**84.88**	**91.39**	92.62
TLOB	81.55	82.68	90.03	**92.81**

7.1 FI-2010 results

Table 2 presents the performance comparison across four prediction horizons[5] for the FI-2010 benchmark dataset. In the Appendix (B) we report also the precision and recall curves for horizon 100. MLPLOB and TLOB exhibit very high precision, also at high recall values, consistently achieve higher precision at all recall levels compared to the other models. The results for the baselines are extracted from the benchmark of Prata et al. [33][6] since the settings are equal for the FI-2010 dataset. MLPLOB and TLOB outperform all the other models analyzed in [33], surpassing state-of-the-art performance. Interestingly, MLPLOB demonstrates the best performance in the first three horizons. Notably, the performance differential between MLPLOB and TLOB is minimal, which, as we will demonstrate in Section 7.2, can be attributed to the lower complexity of the FI-2010 dataset, which explains the uselessness of a more complex architecture such as TLOB for this particular dataset.

7.2 Tesla and Intel results

In Table 3 we show the results for Tesla and in Table 4 for Intel. For each stock, we trained a different model. In the Appendix (B) we report also the precision and recall curves for a horizon equal to 100. For INTC, they exhibit excellent precision at low recall

---

[5]Note that the horizon values represent the number of events before the sampling process of the dataset, while in the benchmarks [31, 33] the values represent the horizons after the sampling process. In other words, the horizons considered are the same and are the ones defined originally in FI-2010.

[6]if we had taken the results reported in the individual papers, MLPLOB and TLOB would have still outperformed all the other models.

Table 3: F1-score for Tesla on four horizons. Bold values show the best scores.

Model	TSLA F1-Score (%) ↑			
	h = 10	h = 20	h = 50	h = 100
DeepLOB	36.25	36.58	35.29	34.43
BiNCTABL	58.69	48.83	42.23	38.77
MLPLOB	**60.72**	**50.25**	38.97	32.95
TLOB	60.50	49.74	**43.48**	**39.84**

Table 4: F1-score for Intel on four horizons. Bold values show the best scores.

Model	INTC F1-Score (%) ↑			
	h = 10	h = 20	h = 50	h = 100
DeepLOB	68.13	63.70	40.3	30.1
BiNCTABL	72.65	66.57	53.99	41.08
MLPLOB	**81.15**	**73.25**	55.74	43.18
TLOB	80.15	72.75	**62.07**	**50.14**

values, indicating their ability to accurately identify the most confident positive instances. MLPLOB outperforms every model on the first two horizons (10, 20), while on the longer horizons (50, 100) TLOB outperforms every model. This is expected since Transformers excels at long-range dependencies. Notably, the difference in performance between MLPLOB and TLOB for the shorter horizons is minimal (≈ 0.5), while on the longer horizons is significant (≈ 7). As expected the longer the horizon the more difficult to forecast. In general, the performances are much lower with respect to FI-2010. We conjecture that this is due to the fact that FI-2010 is characterized by a lower level of complexity with respect to NASDAQ stocks. This derives from the fact that it is composed of Finnish stocks, which are less liquid and efficient than NASDAQ stocks such as Intel and Tesla. Additionally, the data dates back to 2010. Indeed, as will be demonstrated in the subsequent experiment, the prediction difficulty augments as time goes by. The results of our experiment are supported by several works on the topic [33, 37, 48]. All the models are trained until convergence. Notably, both TLOB and MLPLOB achieve convergence in less than half the epochs required by BiNCTABL and DeepLOB.

7.3 Are stocks harder to forecast than in the past?

Table 5: F1-score for Intel on two different periods, from 2012 and 2015. The horizon is set to 50.

Model	F1-Score (%) ↑	
	INTC 2015	INTC 2012
TLOB	60.19	66.87

This experiment examines the challenges associated with market prediction over time and the self-destruction of predictable patterns in financial markets. Empirical evidence consistently demonstrates that forecasting models effective in certain periods become obsolete over time. Several studies indicate that previously observed predictability patterns disappeared after becoming widely known. Dimson and Marsh [15] found this for the UK small-cap premium, while Bossaert and Hillion [2] noted a decline in international stock return predictability around 1990. Aiolfi and Favero [1] reported similar findings for US stocks in the 1990s. The market is increasingly efficient and difficult to predict as time goes by. We extend this investigation to our best-performing model TLOB. Specifically, we tested on a day of Intel from 2012/06/21[7] and confronted the difference in performance with 2015/01/30. We report the performance in Table 5. As expected the performance from 2012 is better than that from 2015. We confirm the hypothesis and the empirical evidence from other works.

7.4 Alternative Threshold Definition Using Average Spread

Table 6: F1-score on Tesla with θ set to the average spread.

Model	F1-Score (%) ↑		
	h = 50	h = 100	h = 200
TLOB	41.39	36.48	30.82

Based on the fact that predictability has to be considered in relation to the transaction costs, we explore an alternative approach to define the trend classification parameter θ, setting it equal to the average spread as a percentage of the mid-price, reflecting the primary transaction cost. This methodology could only be applied to Tesla data, as Intel's higher trading volume (approximately 10 times greater in January 2015) and lower volatility relative to traded shares would result in 99.99% of trends classified as stationary. We set the horizons to 50, 100, and 200 because with shorter horizons 99% of the mid-price movements would be classified as stationary. In Table 6 we report the results. In general, performances show a deterioration, which is probably caused by the classes' unbalance. This experiment highlights the necessity for further refinements in trend definition and method complexity when targeting profitability in practical applications.

7.5 Ablation Study

To evaluate the contribution of each attention mechanism within the TLOB architecture, we performed an ablation study on the FI-2010 dataset. Specifically, we compared the performance of the complete TLOB model against two ablated versions: one without spatial attention (TLOB w/o SA) and another without temporal attention (TLOB w/o TA). To avoid inconsistency, we maintain

the total number of layers fixed[8]. The F1-scores for each model across four prediction horizons (h = 10, 20, 50, and 100) are presented in Table 7. The results demonstrate that the full TLOB model, incorporating both spatial and temporal attention mechanisms, consistently outperforms both ablated versions across all prediction horizons. The performance gain of the full TLOB model highlights the importance of capturing both spatial relationships between LOB features and temporal dependencies across LOB snapshots. This suggests that the dual-attention mechanism effectively learns complementary information, leading to improved predictive accuracy compared to models relying on only one type of attention.

Table 7: Ablation study results. F1-score on the FI-2010 dataset on four horizons. Bold values show the best scores.

Model	FI-2010 F1-Score (%) ↑			
	h = 10	h = 20	h = 50	h = 100
TLOB w/o SA	79.59	78.96	87.51	91.40
TLOB w/o TA	80.27	79.20	87.72	91.42
TLOB	**81.55**	**82.68**	**90.03**	**92.81**

8 Conclusion

We proposed two new deep learning models MLPLOB: A simplified yet effective MLP-based architecture and TLOB, a Transformer-based approach, for the task of stock price trend prediction on Limit Order Book (LOB) data. Both models demonstrated superior performance compared to existing state-of-the-art approaches, with TLOB showing particular promise in handling high-frequency market data. NASDAQ stocks (Tesla, Intel) proved significantly more challenging to predict than Finnish stocks (FI-2010). Our research also showed that prediction accuracy decreases as the forecasting horizon increases, highlighting the inherent challenges of long-term prediction in financial markets.

Limitations: When considering practical implementation, we found that defining trend thresholds based on average spread (transaction costs) significantly impacts model evaluation and potential profitability. This finding underscores the critical gap between academic performance metrics and practical trading applicability.

Future works Looking ahead, several avenues for future research emerge. The investigation of scaling laws for financial deep learning models remains an open question, as does the development of more robust approaches to handling increased market efficiency and complexity. Additionally, the exploration of alternative trend definition methodologies that better align with practical trading constraints could prove fruitful.

Risks: Firstly, it is important to acknowledge that the proposed methodologies are not sufficiently mature for practical deployment in live trading environments. Furthermore, the application of deep learning models to stock price prediction and subsequent utilization in trading carries significant inherent risks. A primary concern is the limited explainability of such models. Furthermore, automated

---

[7]we remark that in a single day of Intel, there are hundreds of thousands of orders making the experiment statistically significant. Furthermore, the trading day was extracted from the LOBSTER public sample files available at https://lobsterdata.com/info/DataSamples.php and it was the only day available, eliminating the possibility of cherry picking.

---

[8]TLOB has 4 temporal attention layers and 4 spatial attention layers, TLOB w/o SA has 8 temporal attention layers and TLOB w/o TA has 8 spatial attention layers

AI models, increasingly integrated into financial markets, present significant risks to financial stability due to their potential to amplify systemic vulnerabilities. These models, often operating with limited transparency, can trigger rapid and widespread market reactions, exacerbating volatility and potentially leading to cascading failures across the financial system.

References

[1] Marco Aiolfi and Carlo A Favero. 2005. Model uncertainty, thick modelling and the predictability of stock returns. *Journal of Forecasting* 24, 4 (2005), 233–254.

[2] Peter Bossaerts and Pierre Hillion. 1999. Implementing statistical criteria to select return forecasting models: what do we learn? *The Review of Financial Studies* 12, 2 (1999), 405–428.

[3] J.P. Bouchaud, J. Bonart, J. Donier, and M. Gould. 2018. *Trades, Quotes and Prices: Financial Markets Under the Microscope*. Cambridge University Press. https://books.google.it/books?id=u45LDwAAQBAJ

[4] Jean-Philippe Bouchaud, J Doyne Farmer, and Fabrizio Lillo. 2009. How markets slowly digest changes in supply and demand. In *Handbook of financial markets: dynamics and evolution*. Elsevier, 57–160.

[5] Jean-Philippe Bouchaud, Marc Mézard, and Marc Potters. 2002. Statistical properties of stock order books: empirical results and models. *Quantitative finance* 2, 4 (2002), 251.

[6] Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165* (2020).

[7] David Byrd, Maria Hybinette, and Tucker Hybinette Balch. 2020. ABIDES: Towards high-fidelity multi-agent market simulation. In *Proceedings of the 2020 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*. 11–22.

[8] Xiangning Chen, Chen Liang, Da Huang, Esteban Real, Kaiyuan Wang, Hieu Pham, Xuanyi Dong, Thang Luong, Cho-Jui Hsieh, Yifeng Lu, et al. 2024. Symbolic discovery of optimization algorithms. *Advances in neural information processing systems* 36 (2024).

[9] Kyunghyun Cho. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[10] Andrea Coletta, Matteo Prata, Michele Conti, Emanuele Mercanti, Novella Bartolini, Aymeric Moulin, Svitlana Vyetrenko, and Tucker Balch. 2021. Towards realistic market simulations: a generative adversarial networks approach. In *Proceedings of the Second ACM International Conference on AI in Finance*. 1–9.

[11] Rama Cont. 2001. Empirical properties of asset returns: stylized facts and statistical issues. *Quantitative finance* 1, 2 (2001), 223.

[12] Rama Cont. 2011. Statistical modeling of high-frequency financial data. *IEEE Signal Processing Magazine* 28, 5 (2011), 16–25.

[13] Rama Cont, Arseniy Kukanov, and Sasha Stoikov. 2014. The price impact of order book events. *Journal of financial econometrics* 12, 1 (2014), 47–88.

[14] P Kingma Diederik. 2014. Adam: A method for stochastic optimization. *(No Title)* (2014).

[15] Elroy Dimson and Paul Marsh. 1999. Murphy's law and market anomalies. *Journal of Portfolio Management* 25, 2 (1999), 53–69.

[16] Martin D Gould, Mason A Porter, Stacy Williams, Mark McDonald, Daniel J Fenn, and Sam D Howison. 2013. Limit order books. *Quantitative Finance* 13, 11 (2013), 1709–1742.

[17] Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415* (2016).

[18] Jonathan Ho, Nal Kalchbrenner, Dirk Weissenborn, and Tim Salimans. 2019. Axial attention in multidimensional transformers. *arXiv preprint arXiv:1912.12180* (2019).

[19] Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. 2022. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848* (2022).

[20] Shi Bin Hoo, Samuel Müller, David Salinas, and Frank Hutter. 2025. The tabular foundation model TabPFN outperforms specialized time series forecasting models based on simple features. *arXiv preprint arXiv:2501.02945* (2025).

[21] Hanna Hultin, Henrik Hult, Alexandre Proutiere, Samuel Samama, and Ala Tarighati. 2023. A generative model of a limit order book using recurrent neural networks. *Quantitative Finance* (2023), 1–28.

[22] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361* (2020).

[23] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. 2022. Transformers in vision: A survey. *ACM computing surveys (CSUR)* 54, 10s (2022), 1–41.

[24] Damian Kisiel and Denise Gorse. 2022. Axial-lob: High-frequency trading with axial attention. In *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 1327–1333.

[25] Robert Kissell. 2020. *Algorithmic trading methods: Applications using advanced statistics, optimization, and machine learning techniques*. Academic Press.

[26] Junyi Li, Xintong Wang, Yaoyang Lin, Arunesh Sinha, and Michael Wellman. 2020. Generating realistic stock market order streams. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 727–734.

[27] Minh-Thang Luong. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025* (2015).

[28] Stefan Nagel. 2021. *Machine learning in asset pricing*. Vol. 1. Princeton University Press.

[29] Peer Nagy, Sascha Frey, Silvia Sapora, Kang Li, Anisoara Calinescu, Stefan Zohren, and Jakob Foerster. 2023. Generative AI for End-to-End Limit Order Book Modelling: A Token-Level Autoregressive Generative Model of Message Flow Using a Deep State Space Network. *arXiv preprint arXiv:2309.00638* (2023).

[30] Paraskevi Nousi, Avraam Tsantekidis, Nikolaos Passalis, Adamantios Ntakaris, Juho Kanniainen, Anastasios Tefas, Moncef Gabbouj, and Alexandros Iosifidis. 2019. Machine learning for forecasting mid-price movements using limit order book data. *Ieee Access* 7 (2019), 64722–64736.

[31] Adamantios Ntakaris, Martin Magris, Juho Kanniainen, Moncef Gabbouj, and Alexandros Iosifidis. [n. d.]. Benchmark Dataset for Mid-Price Forecasting of Limit Order Book Data with Machine Learning Methods. http://urn.fi/urn:nbn:fi:csc-kata20170601153214969115. N/A.

[32] Nikolaos Passalis, Anastasios Tefas, Juho Kanniainen, Moncef Gabbouj, and Alexandros Iosifidis. 2019. Deep adaptive input normalization for time series forecasting. *IEEE transactions on neural networks and learning systems* 31, 9 (2019), 3760–3765.

[33] Matteo Prata, Giuseppe Masi, Leonardo Berti, Viviana Arrigoni, Andrea Coletta, Irene Cannistraci, Svitlana Vyetrenko, Paola Velardi, and Novella Bartolini. 2024. Lob-based deep learning models for stock price trend prediction: a benchmark study. *Artificial Intelligence Review* 57, 5 (2024), 1–45.

[34] Zijian Shi and John Cartlidge. 2022. State dependent parallel neural hawkes process for limit order book event stream prediction and simulation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1607–1615.

[35] Zijian Shi and John Cartlidge. 2023. Neural Stochastic Agent-Based Limit Order Book Simulation: A Hybrid Methodology. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2023, London, United Kingdom, 29 May 2023 - 2 June 2023*, Noa Agmon, Bo An, Alessandro Ricci, and William Yeoh (Eds.). ACM, 2481–2483. https://doi.org/10.5555/3545946.3598974

[36] Justin Sirignano and Rama Cont. 2021. Universal features of price formation in financial markets: perspectives from deep learning. In *Machine learning and AI in finance*. Routledge, 5–15.

[37] Justin A Sirignano. 2019. Deep learning for limit order books. *Quantitative Finance* 19, 4 (2019), 549–570.

[38] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. 2021. Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems* 34 (2021), 24261–24272.

[39] Dat Thanh Tran, Alexandros Iosifidis, Juho Kanniainen, and Moncef Gabbouj. 2018. Temporal attention-augmented bilinear network for financial time-series data analysis. *IEEE transactions on neural networks and learning systems* 30, 5 (2018), 1407–1418.

[40] Dat Thanh Tran, Juho Kanniainen, Moncef Gabbouj, and Alexandros Iosifidis. 2021. Data normalization for bilinear structures in high-frequency financial time-series. In *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 7287–7292.

[41] Avraam Tsantekidis, Nikolaos Passalis, Anastasios Tefas, Juho Kanniainen, Moncef Gabbouj, and Alexandros Iosifidis. 2017. Forecasting stock prices from the limit order book using convolutional neural networks. In *2017 IEEE 19th conference on business informatics (CBI)*, Vol. 1. IEEE, 7–12.

[42] Avraam Tsantekidis, Nikolaos Passalis, Anastasios Tefas, Juho Kanniainen, Moncef Gabbouj, and Alexandros Iosifidis. 2017. Using deep learning to detect price change indications in financial markets. In *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE, 2511–2515.

[43] Avraam Tsantekidis, Nikolaos Passalis, Anastasios Tefas, Juho Kanniainen, Moncef Gabbouj, and Alexandros Iosifidis. 2020. Using deep learning for price prediction by exploiting stationary limit order book features. *Applied Soft Computing* 93 (2020), 106401.

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[45] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. 2022. Transformers in time series: A survey. *arXiv preprint arXiv:2202.07125* (2022).

[46] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. 2023. Are transformers effective for time series forecasting?. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 37. 11121–11128.

[47] Zihao Zhang and Stefan Zohren. 2021. Multi-Horizon Forecasting for Limit Order Books: Novel Deep Learning Approaches and Hardware Acceleration using Intelligent Processing Units. arXiv:2105.10430 [cs.LG] https://arxiv.org/

abs/2105.10430

[48] Zihao Zhang, Stefan Zohren, and Stephen Roberts. 2019. Deeplob: Deep convolutional neural networks for limit order books. *IEEE Transactions on Signal Processing* 67, 11 (2019), 3001–3012.

[49] Zihao Zhang, Stefan Zohren, and Stephen Roberts. 2020. Deep reinforcement learning for trading. *The Journal of Financial Data Science* 2, 2 (2020), 25–40.

A Hyperparameters Search

To find the best hyperparameters, we employ a grid search exploring different values as shown in Table 8. Regarding the hyperparameters of DeepLOB and BiNCTABL, we used the one used in [33] after a large hyperparameters search. We remark that with higher sequence sizes than 128, the performances reach a plateau. For TLOB we searched also for the optimal number of heads, and we noted that there was not difference in between performance between 1, 2, 4 and so we fixed the number of heads to 1.

Table 8: The hyperparameter search spaces and best choices for each model.

Hyperparameter	Search Space	TLOB	MLPLOB
Optimizer	{Adam [14], Lion [8]}	Adam	Adam
Sequence size	{64, 128, 256, 384, 512}	128	128
Learning rate	{0.001, 0.0003, 0.0001}	0.0001	0.003
Number of layers	{2, 3, 4, 6}	4	3

B Additional Results

We report the precision and recall curves for FI-2010 (Fig. 4), INTC (Fig. 5) and TSLA (Fig. 6), for horizon 100. As shown, across the different datasets, TLOB and MLPLOB consistently achieve higher precision at all recall levels compared to the other models. TLOB and MLPLOB, for INTC, exhibit excellent precision at low recall values, indicating their ability to accurately identify the most confident positive instances. Specifically for FI-2010, they exhibit very high precision, also at high recall values.
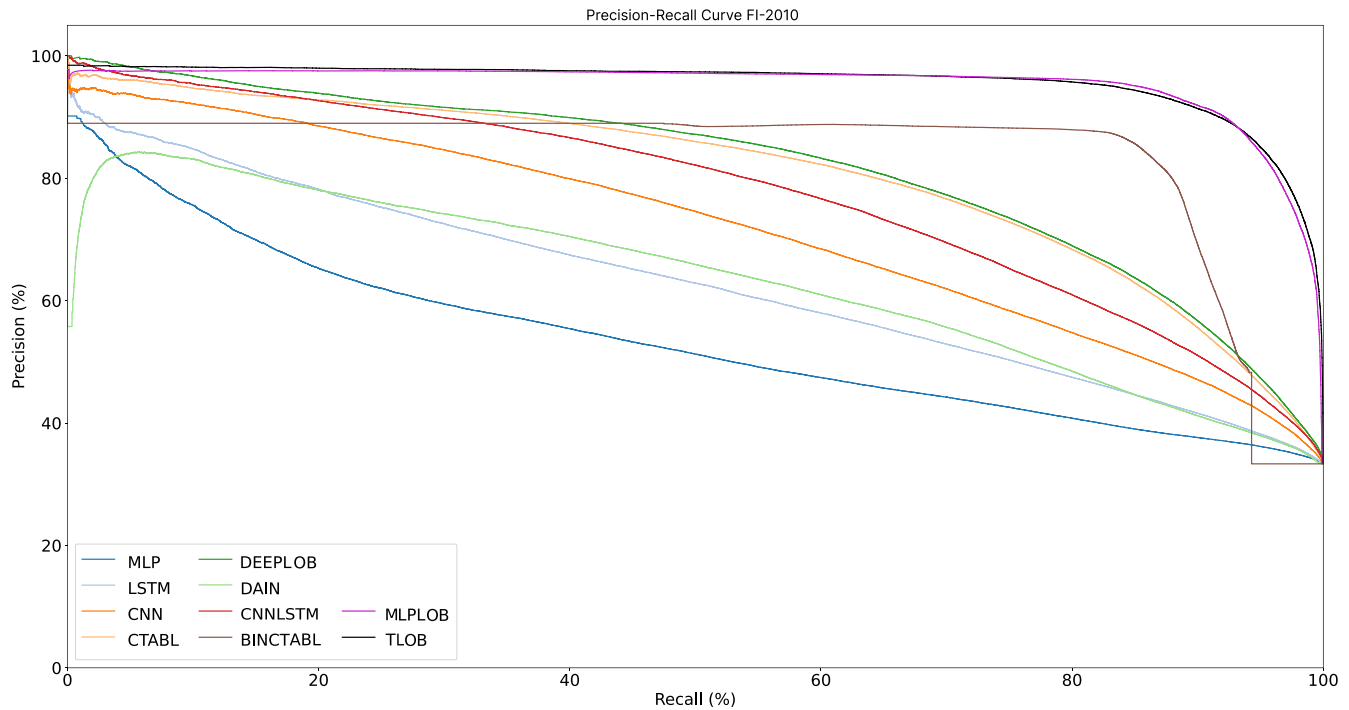
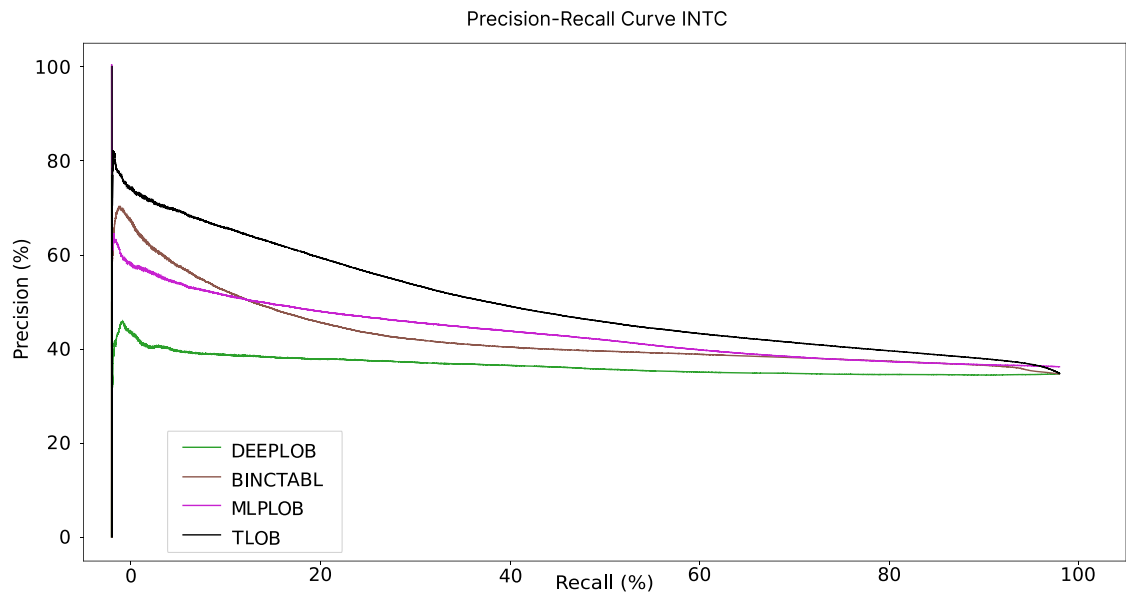**Figure 4: Precision and Recall curve for FI-2010 for horizon = 100.**

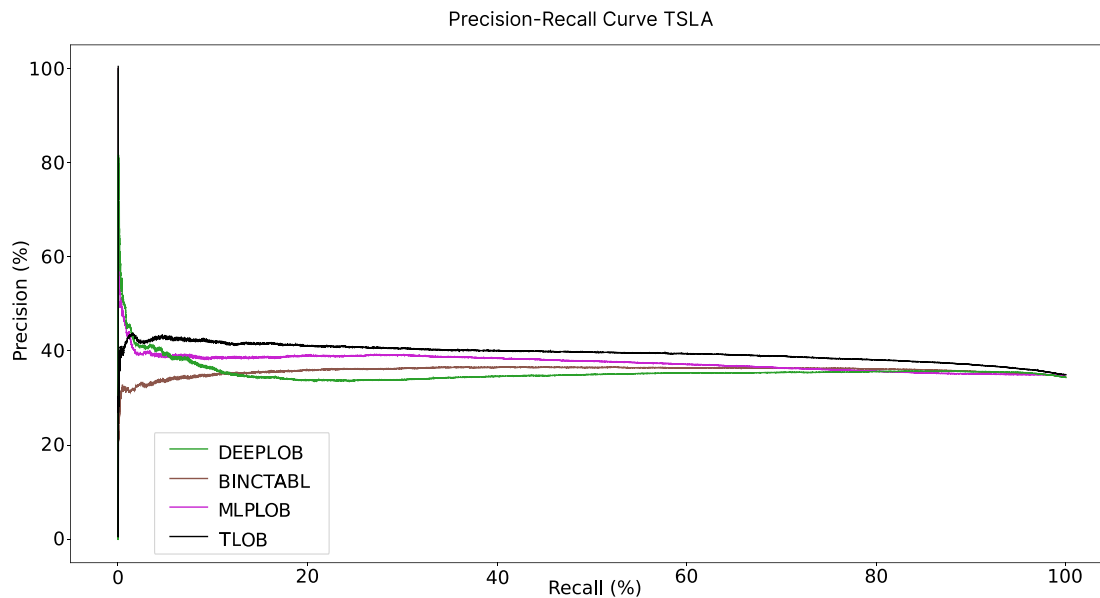**Figure 5: Precision and Recall curve for INTC for horizon = 100.**

Figure 6: Precision and Recall curve for TSLA for horizon = 100.