

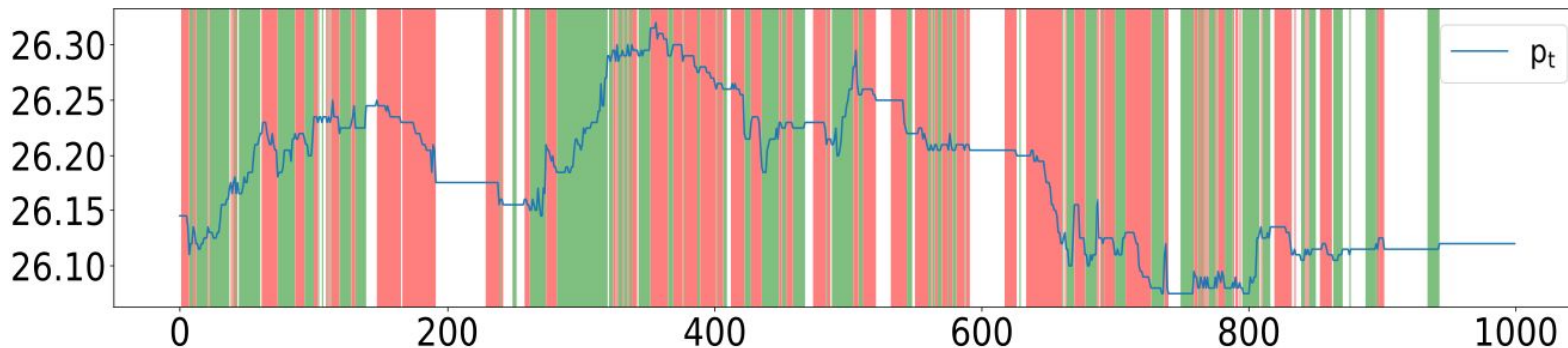
Price movement prediction with DL





Task

Develop a **deep neural network** to **predict future stock price movements** in a large-scale high-frequency **LOB data**.





Limit Order Book (LOB)

The Limit Order Book (LOB) represents a snapshot of the supply and demand for an exchange traded instrument at a given time.

Two types of orders:

- **Bid** orders: orders to **buy** an asset **at or below** a specified price.
- **Ask** orders: orders to **sell** an asset **at or above** a specified price.

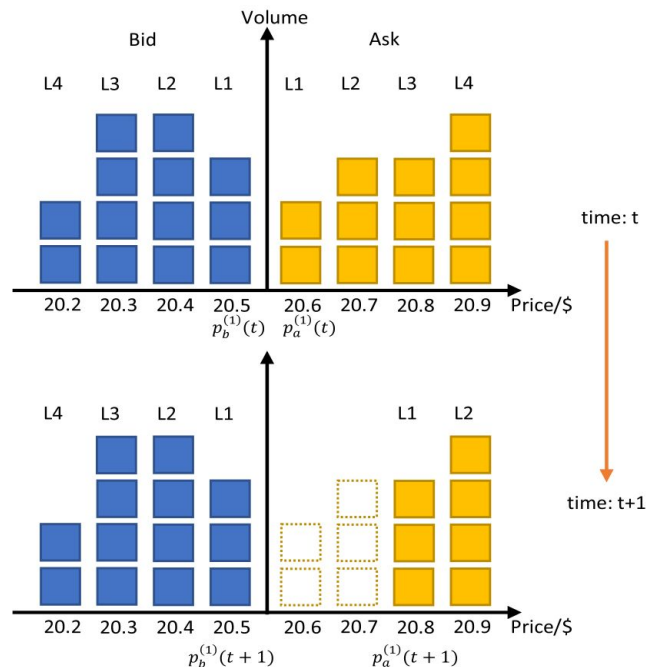
Orders are sorted into different levels based on their submitted prices.



Limit Order Book (LOB)

Both bid and ask orders are represented by a **price** and a **volume**.

The example shows the LOB in two following time steps. The bottom plot shows the action of an incoming market order to buy 5 shares.





Dataset

To train and test our model, we use the **FI-2010 dataset**, limit order data extracted from the Nasdaq Nordic stock market for a time of **10 consecutive days**.

Each state of the LOB contains 10 levels on each side. Therefore, we have a total of 40 features at each timestamp.

- **Training + Validation:** first 7 days (80/20).
- **Testing:** last 3 days.



Input data

We use the 100 most recent states of the LOB as an input to our model:

$$\mathbf{X} = [x_1, x_2, \dots, x_t, \dots, x_{100}]^T \in \mathbb{R}^{100 \times 40}$$

where

$$x_t = [p_a^{(i)}(t), v_a^{(i)}(t), p_b^{(i)}(t), v_b^{(i)}(t)]_{i=1}^{n=10}$$

and p and v respectively represent the price and volume.



Normalization

The performance of machine learning algorithms often depends on how the data is normalized.

Three possibilities in the FI-2010 dataset:

1. **Z-score normalization:** $x' = \frac{x - \mu}{\sigma}$
2. **Min-Max normalization:** $x' = \frac{x - \min(x)}{\max(x) - \min(x)}$
3. **Decimal precision normalization:** $x' = \frac{x}{10^k}, \quad k = \min_k \{k \mid \max(|x|) < 1\}$



Labelling

To create **labels that represent the direction of price changes**, we use the **mid-price**:

$$p_t = \frac{p_a^{(1)}(t) + p_b^{(1)}(t)}{2}$$

There are two strategies to compute labels:

- **Strict**
- **Smooth**



Labelling - Strict **✗**

We compare the mid-prices at two following time steps:

$$p_{t+1} > p_t \Rightarrow \nearrow$$

$$p_{t+1} = p_t \Rightarrow \rightarrow$$

$$p_{t+1} < p_t \Rightarrow \searrow$$

Problem: financial data is highly stochastic, thus the label set would be noisy.



Labelling - Smooth¹ ✓

We use the mean of the next k mid-prices, and we compare the percentage change against a threshold α to compute the labels:

$$m_+(t) = \frac{1}{k} \sum_{i=0}^k p_{t+i}$$

$$l_t > \alpha \Rightarrow \nearrow$$

$$-\alpha < l_t < \alpha \Rightarrow \rightarrow$$

$$l_t = \frac{m_+(t) - p_t}{p_t}$$

$$l_t < -\alpha \Rightarrow \searrow$$

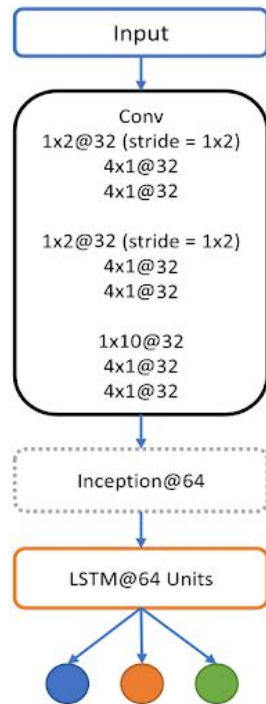
1: In the FI-2010 dataset, the data is already labelled according to the smooth labeling technique.



Model Architecture

The network architecture comprises three building blocks:

1. **Convolutional Modules.**
2. **Inception Module.**
3. **LSTM Module + Classifier.**





Convolutional Modules

Input size = $(1, 100, 40)$.

1. $K = (1, 2), S = (1, 2)$: the **first block** summarises **information between price and volume** at each order book level.
2. $K = (1, 2), S = (1, 2)$: the **second block** integrates **information across bid and ask orders**.
3. $K = (1, 10)$: the **third block** integrates **all information across multiple order book levels** by using a large filter.

Output size = $(32, 100, 1)$.

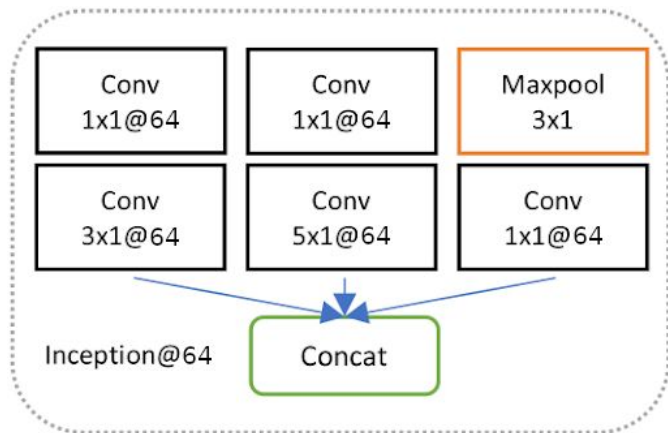


Inception Module

Convolutional filters of size $(k, 1)$ can be used to capture local interactions amongst data over k time steps.

The **Inception Module** is used to capture **dynamic behaviour over multiple timescales**, wrapping several convolutions together.

Output size = $(194, 100, 1)$.





LSTM Module + Classifier

The **LSTM Module** is used to **capture temporal relationships** that exists in the extracted features.

The **final classifier block** uses a **fully connected layer** with a **softmax activation**, and hence the final output elements represent the **probability of each price movement class** at each time step.

Output size = $(1, 3)$.



Hyperparameters

Input window length	100
Projection horizon	10
LOB levels	10
Learning rate	0.0001
Batch size	128
Early stopping	ON
Patience	20



References

[1]: A. Ntakaris, M. Magris, J. Kannianen, M. Gabbouj, and A. Iosifidis, “[Benchmark dataset for mid-price prediction of limit order book data with machine learning methods](#)”, J. Forecasting, vol. 37, no. 8, pp. 852–866, 2018.

[2]: M. D. Gould, M. A. Porter, S. Williams, M. McDonald, D. J. Fenn, and S. D. Howison, “[Limit order books](#)”, Quantitative Finance, vol. 13, no. 11, pp. 1709–1742, 2013.

[3]: Z. Zhang, S. Zohren, and S. Roberts, “[DeepLOB: Deep Convolutional Neural Networks for Limit Order Books](#)”, IEEE Transactions on Signal Processing, vol. 67, no. 11, pp. 3001–3012, 2019.