Long Short Trader Using Deep Learning

PO-237

Luiza Santos

ITA, Operations Research

November 13, 2024

Agenda

Agenda

1. Motivation and goals
2. Data ingestion
3. Data processing and feature engineering
4. Model Architecture
5. Trading Strategy
6. Simulation Results
7. Next steps

Agenda

Agenda

1. Motivation and goals
2. Data ingestion
3. Data processing and feature engineering
4. Model Architecture
5. Trading Strategy
6. Simulation Results
7. Next steps

Agenda

1. Motivation and goals
2. Data ingestion
3. Data processing and feature engineering
4. Model Architecture
5. Trading Strategy
6. Simulation Results
7. Next steps

Agenda

1. Motivation and goals
2. Data ingestion
3. Data processing and feature engineering
4. Model Architecture
5. Trading Strategy
6. Simulation Results
7. Next steps

Agenda

Motivation and Goals

Project Motivation

- Leverage deep learning frameworks to tackle the challenges of time series data in financial markets.
- Utilize a portfolio allocation strategy to maximize profits

- Leverage deep learning frameworks to tackle the challenges of time series data in financial markets.
- Utilize a portfolio allocation strategy to maximize profits

Motivation and Goals

Challenges and Learning

- Time Series Complexity: Adapting LSTMs for time series analysis, which included handling dependencies and trends.
- Training using Sliding Window: Mastering the sliding window approach to split data effectively for prediction intervals on a large, real-world dataset.
- New Deep Learning Framework: Learning PyTorch from scratch to implement LSTM models, which was both challenging and rewarding due to its flexibility and control over model architecture.

- Time Series Complexity: Adapting LSTMs for time series analysis, which included handling dependencies and trends.
- Training using Sliding Window: Mastering the sliding window approach to split data effectively for prediction intervals on a large, real-world dataset.
- New Deep Learning Framework: Learning PyTorch from scratch to implement LSTM models, which was both challenging and rewarding due to its flexibility and control over model architecture.

Motivation and Goals

Challenges and Learning

- Time Series Complexity: Adapting LSTMs for time series analysis, which included handling dependencies and trends.
- Training using Sliding Window: Mastering the sliding window approach to split data effectively for prediction intervals on a large, real-world dataset.
- New Deep Learning Framework: Learning PyTorch from scratch to implement LSTM models, which was both challenging and rewarding due to its flexibility and control over model architecture.

Process Flow

Overview

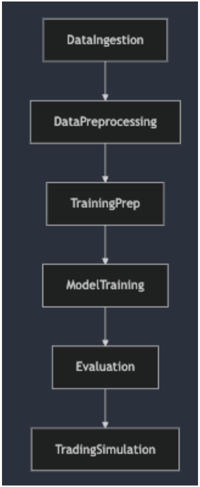


Figure: Process Flow

Data Ingestion

COMDINHEIRO API

Brazilian Stock Market

- Market data
- Fundamentalist Data

Total of 26 features

Data processing and feature engineering

- Challenges of data serialization and memory constraints
  - Solution: batch training
  - Stocks traded in parallel per time window
  - Model weights are updated after each time-window
- Creating target variable: return direction

$$T = \mathbb{1}(P_t - P_{t-1} > 0)$$

- Dealing with missing values: forward fill
- Validation with sliding window

Data processing and feature engineering

- Challenges of data serialization and memory constraints
  - Solution: batch training
  - Stocks traded in parallel per time window
  - Model weights are updated after each time-window
- Creating target variable: return direction

$$T = \mathbb{1}(P_t - P_{t-1} > 0)$$

- Dealing with missing values: forward fill
- Validation with sliding window

Data processing and feature engineering

- Challenges of data serialization and memory constraints
  - Solution: batch training
  - Stocks traded in parallel per time window
  - Model weights are updated after each time-window
- Creating target variable: return direction

$$T = \mathbb{1}(P_t - P_{t-1} > 0)$$

- Dealing with missing values: forward fill
- Validation with sliding window

Data processing and feature engineering

- Challenges of data serialization and memory constraints
  - Solution: batch training
  - Stocks traded in parallel per time window
  - Model weights are updated after each time-window
- Creating target variable: return direction

$$T = \mathbb{1}(P_t - P_{t-1} > 0)$$

- Dealing with missing values: forward fill
- Validation with sliding window

Data processing and feature engineering

- Challenges of data serialization and memory constraints
  - Solution: batch training
  - Stocks traded in parallel per time window
  - Model weights are updated after each time-window
- Creating target variable: return direction

$$T = \mathbb{1}(P_t - P_{t-1} > 0)$$

- Dealing with missing values: forward fill
- Validation with sliding window

Data processing and feature engineering

- Challenges of data serialization and memory constraints
  - Solution: batch training
  - Stocks traded in parallel per time window
  - Model weights are updated after each time-window
- Creating target variable: return direction

$$T = \mathbb{1}(P_t - P_{t-1} > 0)$$

- Dealing with missing values: forward fill
- Validation with sliding window

Data processing and feature engineering

- Challenges of data serialization and memory constraints
  - Solution: batch training
  - Stocks traded in parallel per time window
  - Model weights are updated after each time-window
- Creating target variable: return direction

$$T = \mathbb{1}(P_t - P_{t-1} > 0)$$

- Dealing with missing values: forward fill
- Validation with sliding window

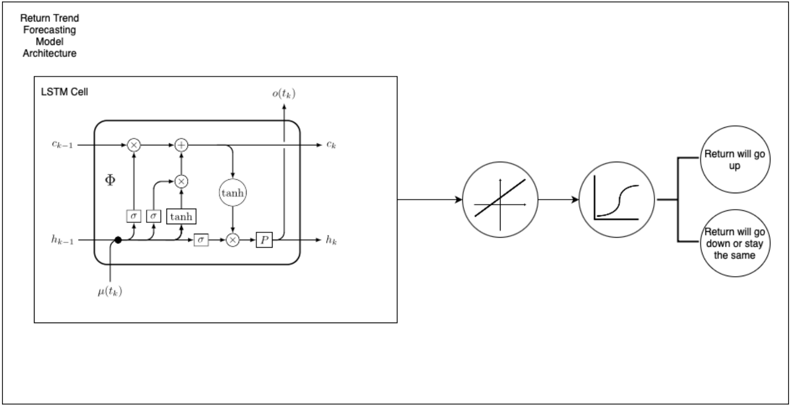


Figure: Model Architecture

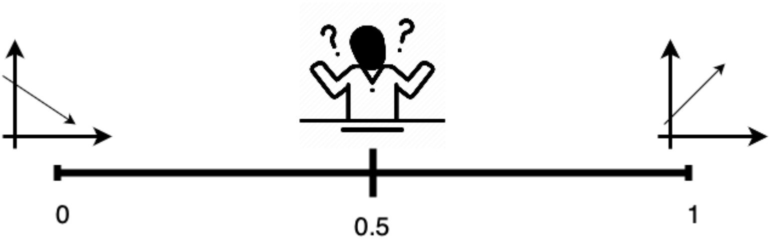


Figure: Model Output

Trading Strategy
Long Short Strategy

- New portfolio daily
- Long strategy: Choose 10 stocks with greater certainty of growth
- Short strategy: Choose 10 stocks with greater certainty of decrease
- Equal weight of the 20 stocks

Trading Strategy

Long Short Strategy

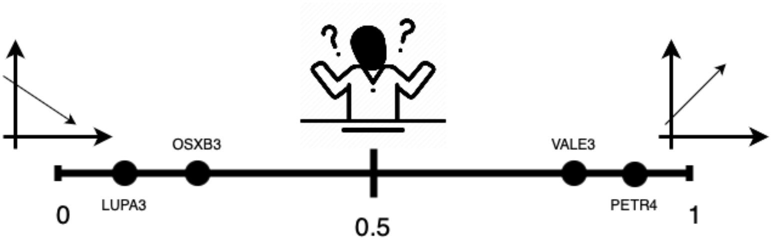
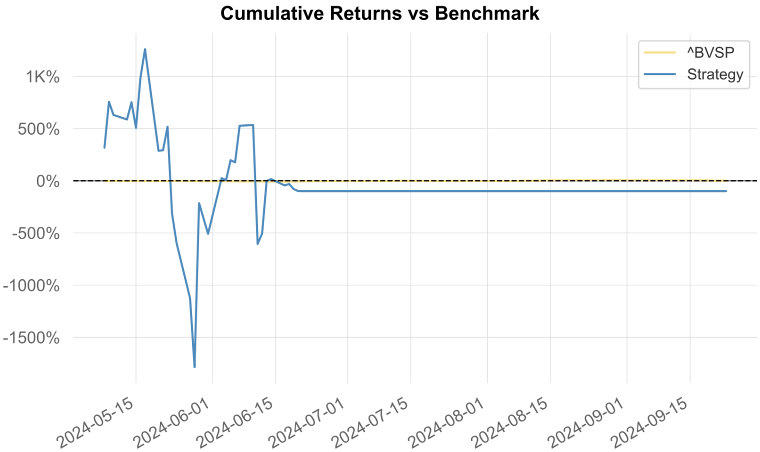


Figure: Model Output

Context: simulated over 100 trading days



Simulation Results

Cumulative Returns Volatility Matched

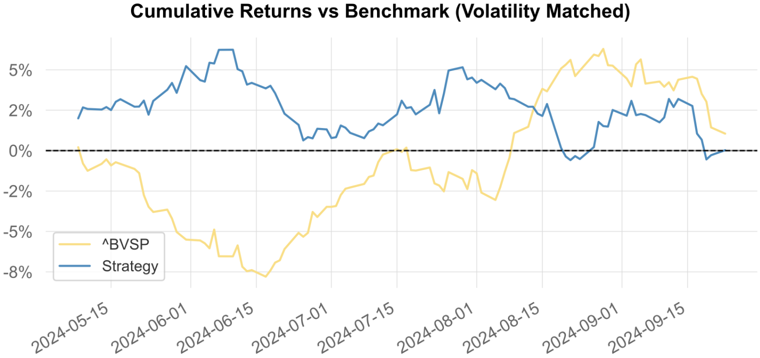


Figure: Cumulative Returns Volatility Matched

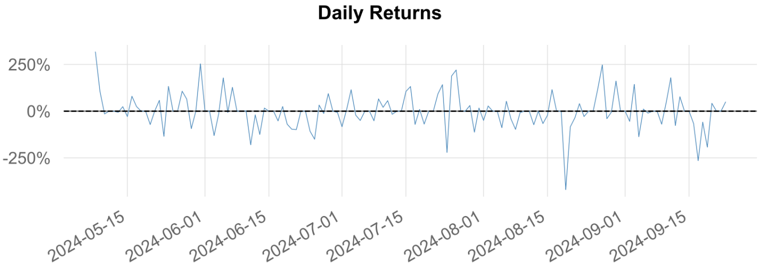


Figure: Daily Returns

Next Steps

Attempt:

- A different model architecture
- More feature engineering
- A different retraining frequency

References

Time-Series-Based Stock Market Analysis using Machine Learning.
Amruth, S. Jaya and Nigelesh, T.M. and Shruthik, V. Sai and Reddy, Valluru Sateesh and Venugopalan, Manju
15th International Conference on Computing Communication and Networking Technologies, 2024.

Multivariate LSTM-FCNs for time series classification
Fazle Karim and Somshubra Majumdar and Houshang Darabi and Samuel Harford
Neural Networks. 166, 2019.

Thank you for listening!

Luiza Santos

luizasantos@ita.br