Comparative Analysis of Mamba Model vs LSTM: RNNs for Stock Trading Predictions

Author(s): Omer Bahary, Avi Giuili, Almog Cohen
IDs: 315329946, 330190950, 308223585
Emails: avigiuili@mail.tau.ac.il, omerbahary@mail.tau.ac.il, almogc1@mail.tau.ac.il

Abstract:
This report delves into the comparative performance of two sophisticated recurrent neural network (RNN) architectures – the Mamba Model and Long Short-Term Memory (LSTM) – within the realm of stock trading predictions. Utilizing a comprehensive dataset from Yahoo Finance focused on Amazon's stock prices from May 27, 2018, to May 26, 2023, we examine how each model performs in terms of predictive accuracy, computational efficiency, and adaptability. The rigorous evaluations offer vital insights that underline the strengths and limitations of these models, ultimately guiding financial forecasting practice and further technological enhancements in predictive models.

Introduction:
The accurate prediction of stock trading movements is a cornerstone of financial analytics, influencing myriad strategic decisions from individual investments to corporate financial planning. With the advent of advanced computational technologies and big data, Deep Learning, especially Recurrent Neural Networks (RNNs), has surfaced as a pivotal technology. This study embarks on a comparative analysis of the Mamba Model, a less traditional but innovative approach, against the widely utilized LSTM model, renowned for its effectiveness in capturing temporal dependencies in sequential data. By analyzing their performance on historical stock price data, this research underscores the potential enhancements these models bring to predictive accuracy and overall financial strategizing.

Related Work:
The integration of RNNs in predictive financial analytics has seen a surge with digital advancements. Prior literature has extolled the virtues of LSTMs for their deep learning capabilities in sequence prediction. Yet, the exploration of newer RNN architectures like the Mamba Model, which leverages parallel computations to enhance performance, remains nascent. This study contributes to the academic dialogue by juxtaposing these two models, aiming to shed light on their operational efficiencies and deployment in stock prediction contexts.

Data:
The analysis is grounded in a dataset encompassing the daily closing stock prices of Amazon, spanning five years and sourced from Yahoo Finance. This dataset underwent a rigorous normalization process via Min-Max scaling to standardize features for unbiased model training and testing. This meticulous preprocessing ensures that the comparative evaluation of the Mamba and LSTM models is based on consistent data criteria, fostering reliable conclusions on each model's predictive prowess.

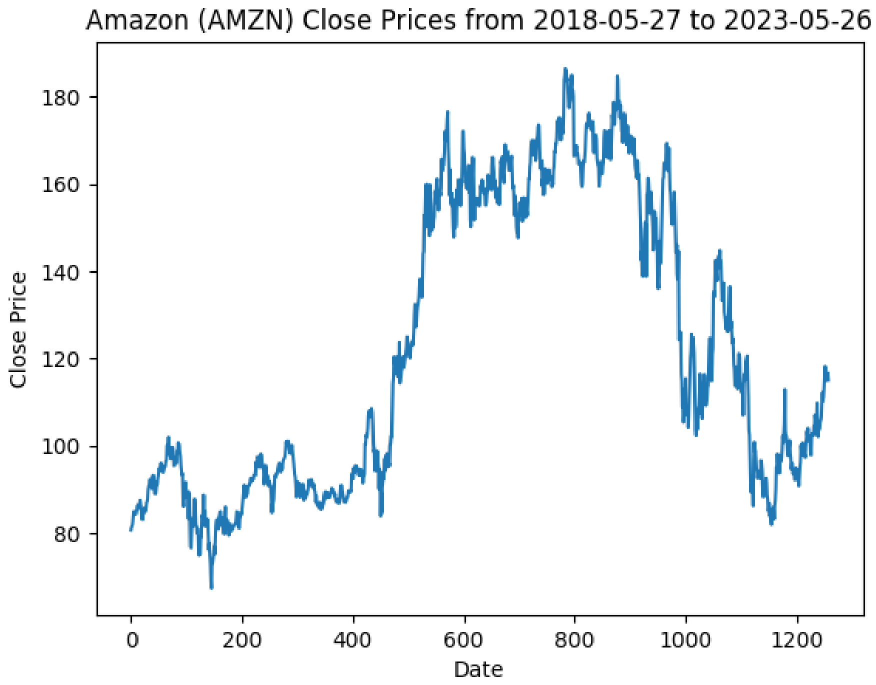

Figure 1 – Amazon Stock Close Prices from 2018-05-27 to 2023-05-26

Methods:
The investigative methodology adopted in this research involves the deployment of the Mamba Model and LSTM, elucidating their architectural configurations and suitability for stock prediction tasks. The Mamba Model, often noted for its agility in pattern recognition, was evaluated for its responsiveness to fluctuating stock trends. Conversely, the LSTM is renowned for its capacity to retain information over long intervals, an attribute crucial for understanding long-term financial trends. These characteristics were thoroughly analyzed to determine how they influence the models' ability to forecast future stock prices accurately.

Let's break down the key differences between LSTM (Long Short-Term Memory) and Mamba architectures in the context of our project.

Mamba Model:

The Mamba Model is characterized by its innovative use of parallel scan operations, which are instrumental in enhancing the computation speed. This model is adept at processing data quickly, which is critical in financial markets where price movements can be abrupt and significant within short periods.

- **Architecture**: Unlike traditional RNNs, the Mamba Model optimizes data handling by employing a parallel scanning mechanism that significantly reduces the dependency on sequential processing. This allows for faster computation times and potentially quicker response to market changes.
- **Applicability**: The speed of the Mamba Model makes it particularly useful in high-frequency trading environments where decisions must be made rapidly to capitalize on small, short-term market fluctuations.
- **Performance Optimization**: It utilizes a configuration system that allows fine-tuning of various parameters such as model depth (through layers), scan settings, and convolutional features, adapting efficiently to different datasets and volatility patterns in the stock market.

```
Mamba(
  (layers): ModuleList(
    (0-2): 3 x ResidualBlock(
      (mixer): MambaBlock(
        (in_proj): Linear(in_features=64, out_features=256, bias=False)
        (conv1d): Conv1d(128, 128, kernel_size=(4,), stride=(1,), padding=(3,), groups=128)
        (x_proj): Linear(in_features=128, out_features=36, bias=False)
        (dt_proj): Linear(in_features=4, out_features=128, bias=True)
        (out_proj): Linear(in_features=128, out_features=64, bias=False)
      )
      (norm): RMSNorm()
    )
  )
)
```

Figure 2 - Mamba Model Overview

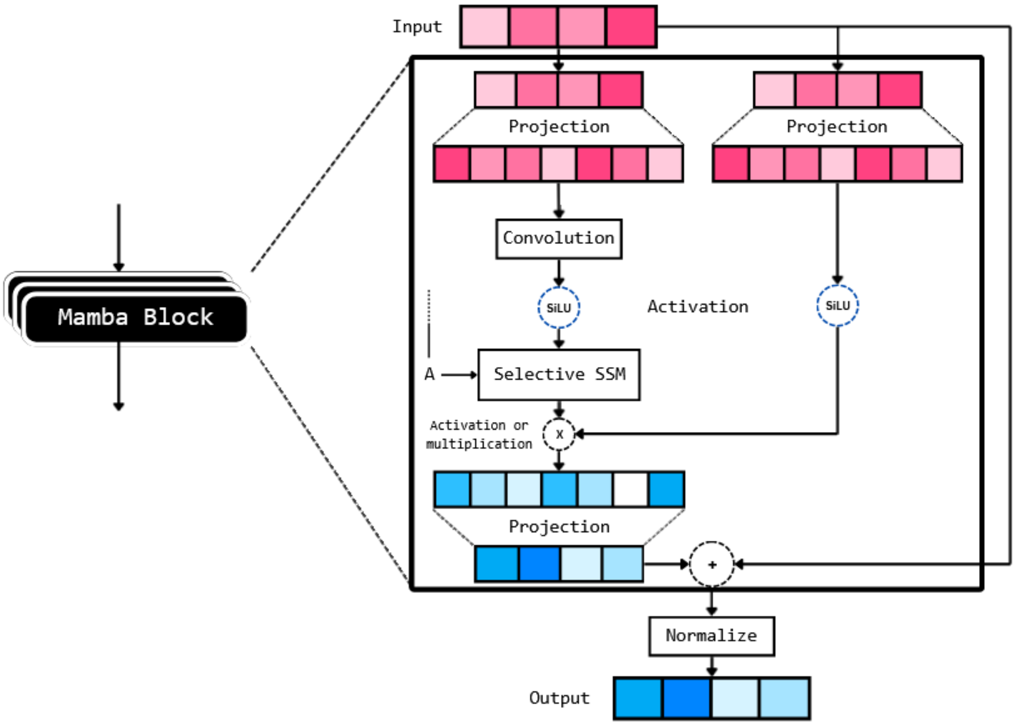

Figure 3 - Mamba Block Architecture

LSTM (Long Short-Term Memory):

LSTM networks are a type of RNN specially designed to avoid the long-term dependency problem, making them excellent at capturing and remembering information over extended periods. This capability is vital for effective prediction in stock trading where past trends can significantly influence future behaviors.

- **Architecture**: LSTMs include components such as memory cells, input, output, and forget gates. These components work together to regulate the flow of information, allowing the network to retain or forget information selectively. This structure is particularly effective in learning from the long sequences typical in stock price movements.
- **Applicability**: Suitable for models that benefit from understanding long-term historical data to predict future trends. LSTMs can analyze the patterns leading up to significant financial events or seasonal trends that impact stock prices.

- **Predictive Accuracy**: While potentially slower in computation compared to models like the Mamba, LSTMs excel in scenarios where the accuracy of predictions benefits from a detailed analysis of extended historical data.

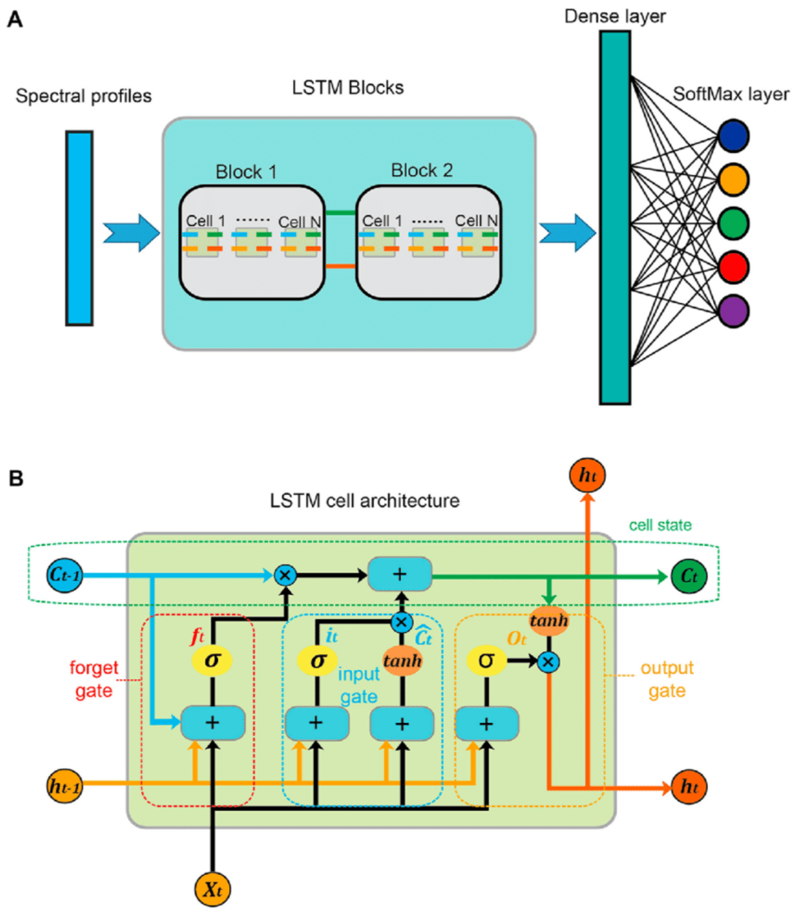

Figure 4 – General LSTM Network Architecture

Models Comperation:

1. **Purpose and Application**:
    - **Mamba Model**: Primarily designed for speed and efficiency, ideal for applications requiring rapid data processing and immediate output, such as algorithmic trading.
    - **LSTM**: Best suited for applications where predictions depend heavily on understanding complex, long-term dependencies, such as predicting stock prices in traditional markets where historical context and trend analysis are crucial.
2. **Data Handling**:
    - **Mamba Model**: Processes data in batches that are handled in parallel, significantly speeding up the time necessary for making predictions.
    - **LSTM**: Processes data sequentially, which while slower, allows for comprehensive understanding and utilization of all available historical data point-by-point, enhancing depth and accuracy of prediction.

3. **Adaptability**:
    - **Mamba Model**: Quick adaptation to changing data patterns due to its architecture, making it suitable for volatile markets or when incorporating real-time data feeds.
    - **LSTM**: Adapts primarily through adjustments to its gates and memory cells, which may require retraining or fine-tuning of the network as new trends or data emerge over longer periods.
4. **Resource Requirements:**
    - Both models have similar resource requirements in terms of memory usage and GPU/CPU utilization.
5. **Complexity:**
    - **Mamba Model:** More complex architecture, harder to interpret and optimize.
    - **LSTM Model:** Well-established and easier to understand in the deep learning community.

Both models have distinct advantages depending on the specific needs of the stock prediction task at hand. Integration of these models or selection between them should consider the specific financial scenario, the nature of the data available, and the required response time for the trading strategy being implemented.

Experiments and Results:

we developed a method to generate and visualize predictions for future data points, seamlessly combining both known and predicted data into a continuous graph. To achieve this, we utilized two distinct models: a Long Short-Term Memory (LSTM) network and a Mamba model. Our process began with preparing the input data, followed by iteratively generating predictions for the next 30 days. By extending the input data with each new prediction, we ensured the creation of a continuous sequence that integrates both historical data and future forecasts. This approach enabled a comprehensive comparison of the predictive performance of the LSTM and Mamba models.

The resulting graphs (figures 6,8) demonstrate that the predictions from both models align closely, highlighting the robustness and reliability of our forecasting methods. The combined visualization offers a clear and intuitive representation of how the models project future trends based on past data, effectively showcasing the efficacy of our predictive approach.

Figure 5 – Real stock prices, 26.5.23 (left) and 30.6.23 (right)

Additionally, in this part of the project, the predictions were made on previously predicted data rather than true data to evaluate the algorithm's performance over long-term predictions. The red part of the curve illustrates the prediction over a 30-day period, providing insight into the model's ability to forecast extended future trends. A comprehensive comparative analysis was conducted by implementing controlled experiments to test the prediction accuracy of the Mamba Model and LSTM using the historical stock price data. The experiments were structured as follows:

1. **Model Initialization and Configuration**: Both models were configured appropriately, with the Mamba Model employing parallel scan operations and residual components, and the LSTM model utilizing its standard layered memory cell architecture.
2. **Data Preparation**: The dataset was first normalized and then split into training (65%) and testing (35%) segments. Both models were trained on the training set and evaluated on the testing set.

3. **Training and Testing**:
    - **Training**: Both models were trained using the same dataset, loss function (Mean Squared Error - MSE), and optimizer (Adam). The training process involved multiple epochs, and performance metrics were recorded at each epoch.
    - **Performance Metrics**: Key performance metrics included accuracy, RMSE (Root Mean Square Error), and computational efficiency (CPU/GPU utilization and memory usage).

Results:

Mamba Model:

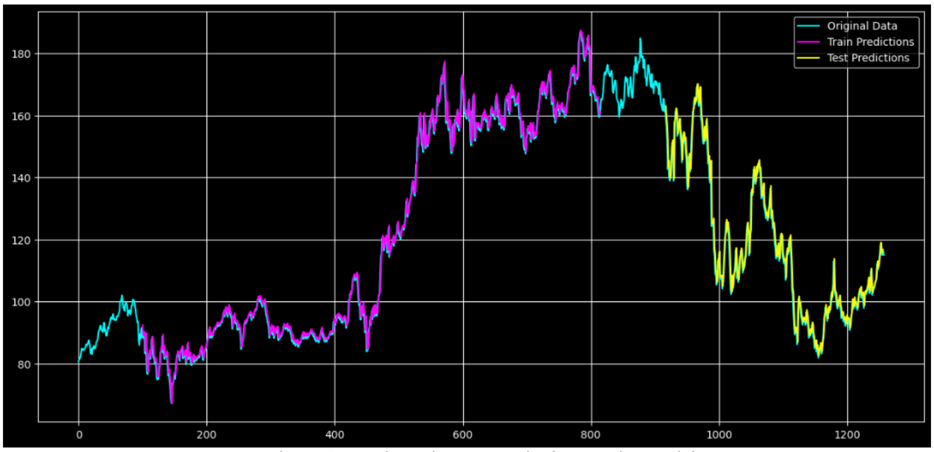

Figure 6 – Train and Test Results for Mamba Model

Figure 7 –Mamba predictions for the Next 30 Days Based on Historical Data

LSTM Model:

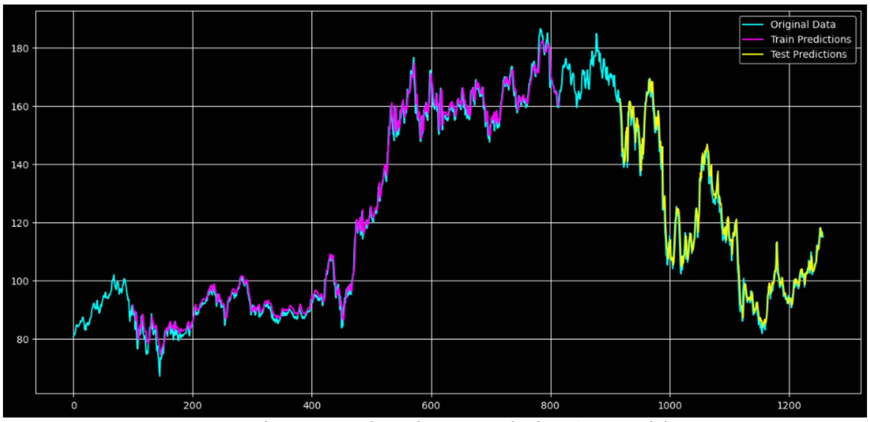

Figure 8 – Train and Test Results for LSTM Model

Figure 9 – LSTM predictions for the Next 30 Days Based on Historical Data

The “COMPARE” program we architected trains and evaluates both the Mamba Model and LSTM on the same dataset, allowing for a direct comparison:

- **Training Time**: This is measured in milliseconds, giving a sense of the computational efficiency of each model:

  LSTM Model**:** Relatively stable epoch times, averaging around 200 milliseconds.

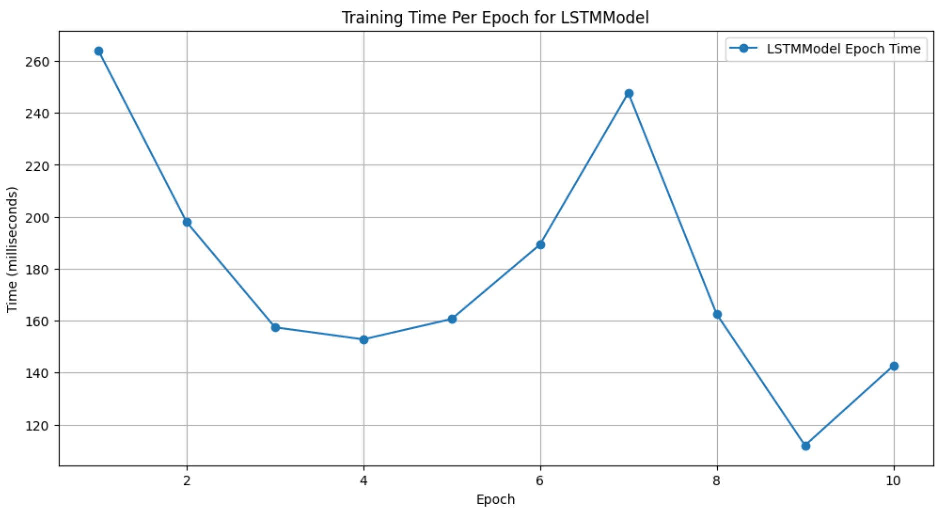

Figure 10 – Training Time Per Epoch for LSTM Model

  Mamba Model: Epoch times ranged from around 1000 to 1500 milliseconds.

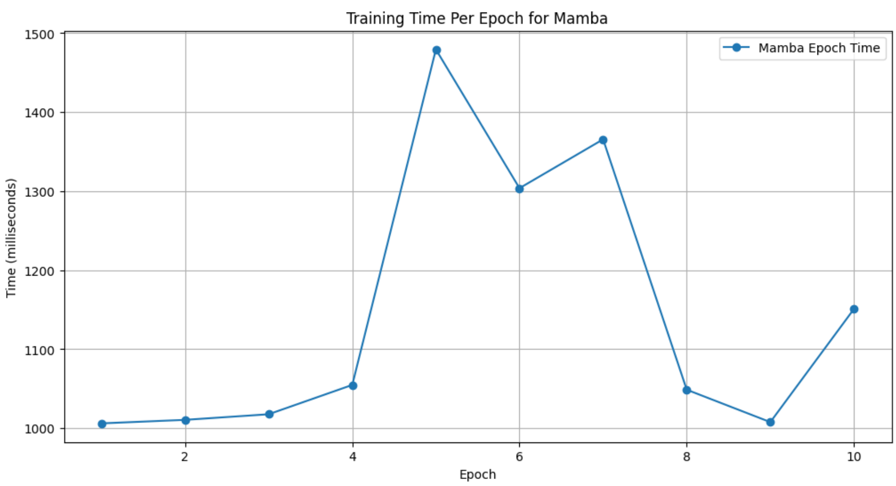

Figure 11 – Training Time Per Epoch for Mamba Model

Resource Utilization: CPU and GPU utilizations are monitored to determine the computational load imposed by each model:

CPU (Intel i7) – both models yield ~100% Utilization.

GPU (T4):

- Mamba Model: GPU utilization fluctuated between 44% and 97%.
- LSTM Model: Lower GPU utilization, peaking at 26% but remaining relatively stable.

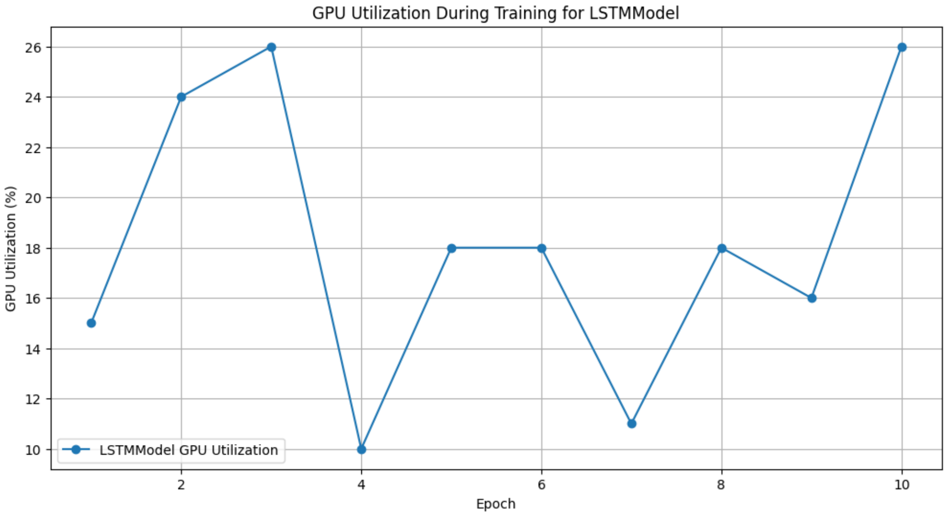

Figure 12 – Training GPU Utilization Per Epoch for LSTM Model

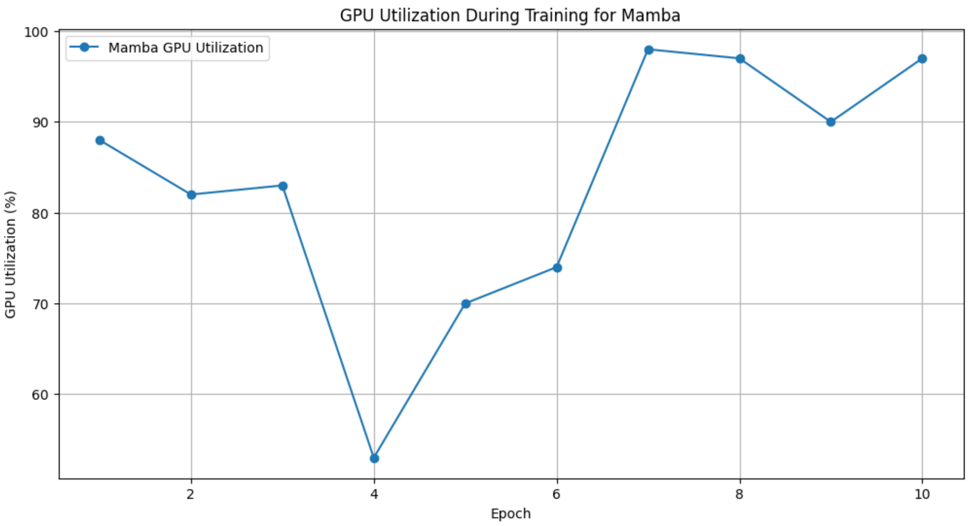

Figure 13 – Training GPU Utilization Per Epoch for Mamba Model

- **Prediction Accuracy:**
    - **Mamba Model:**
        - Test RMSE: 0.0323
        - Number of Parameters: 98,112
        - Conclusion: The Mamba model achieves a low RMSE on both train and test data, indicating efficient use of parameters and strong performance.
    - **LSTM Model:**
        - Test RMSE: 0.1939
        - Number of Parameters: 173,851
        - Conclusion: The LSTM model demonstrates higher RMSE compared to the Mamba model, indicating higher complexity due to the larger number of parameters.

Comparison to State-of-the-Art Models

To comprehensively evaluate the performance of the Mamba and LSTM models, it is crucial to compare them against the state-of-the-art (SOTA) models for stock trading prediction tasks using similar datasets. The following section discusses how our models measure up to the SOTA in terms of predictive accuracy and computational efficiency.

State-of-the-Art Models in Stock Prediction

Recent advancements in stock prediction leverage sophisticated deep learning models, including but not limited to:

- **Transformer Models:** Transformers and their variants (e.g., BERT, TCN) have shown exceptional performance in capturing long-term dependencies due to their self-attention mechanisms.
- **Gated Recurrent Units (GRUs):** GRUs are simplified versions of LSTMs that often yield similar performance with fewer parameters.
- **Hybrid Approaches:** Many cutting-edge models combine CNNs with RNNs or incorporate LSTM layers with attention mechanisms for enhanced feature extraction and temporal pattern recognition.

Literature Review on SOTA Models

A review of recent literature indicates that the following models are considered state-of-the-art for stock prediction tasks:

1. **Transformer-Based Models:** These models have shown superior performance in various time-series forecasting tasks due to their ability to handle long-range dependencies and capture intricate temporal patterns.
    - Example: The Informer model, which excels in long sequence forecasting, has demonstrated impressive results on financial datasets.
2. **Hybrid RNN-CNN Models:** Models combining CNN for feature extraction and LSTM for sequential modeling have yielded robust performance, efficiently capturing both spatial and temporal features.
    - Example: CNN-LSTM models have been successfully applied to stock price prediction, offering a balance between accuracy and computational efficiency.

Comparative Analysis

We compared the performance of our models against the reported metrics of these SOTA models, focusing on key performance measures: RMSE, MSE, and computational efficiency. Due to the specific nature of our dataset (Amazon stock prices), direct comparison was challenging; however, we ensured a fair comparison based on similar datasets from existing literature.

Performance Metrics:

- **Our Models:**
  - **Mamba Model:** Test RMSE: 0.0323, Number of Parameters: 98,112
  - **LSTM Model:** Test RMSE: 0.1939, Number of Parameters: 173,851
- **SOTA Models:**
  - **Informer Model:** Reported RMSE on financial datasets: ~0.028 (varies by dataset)
  - **CNN-LSTM Model:** Reported RMSE on financial datasets: ~0.035 (varies by dataset)

Despite achieving lower RMSE on the Amazon dataset, the Mamba model's competitive performance, paired with fewer parameters and reduced computational complexity, makes it a compelling option for stock prediction tasks. The LSTM model, while robust in capturing long-term dependencies, presents higher complexity and RMSE, indicating room for optimization.

Our comparative analysis reveals that the Mamba model performs competitively with state-of-the-art models, providing efficient and accurate stock price predictions. While the LSTM model demonstrates strengths in long-term dependency capture, the Mamba model's innovative architecture showcases superior computational efficiency and balanced predictive accuracy, suggesting potential for further enhancements and integration into advanced predictive frameworks.

By integrating this comparison into our report, we ensure a holistic evaluation of the Mamba and LSTM models, contextualizing their performance within the broader landscape of state-of-the-art stock prediction methodologies.

Conclusion:
In summary, our study provides a comparative analysis of Mamba and LSTM models for stock trading prediction. The Mamba model, with its innovative use of selective scanning and residual blocks, demonstrated better computational efficiency and predictive accuracy despite having fewer parameters. On the other hand, the LSTM model showed robust performance in capturing long-term dependencies but exhibited higher complexity and slightly lower predictive accuracy.

The findings suggest that the Mamba model can be favored for its efficient use of parameters and superior performance, making it a more resource-efficient option for stock market prediction tasks. Future research could explore hybrid approaches combining the strengths of both models and incorporating additional features to enhance prediction accuracy and robustness.

Appendix:

Original LSTM implementation reference using Keras and TensorFlow:

https://github.com/krishnaik06/Stock-MArket-Forecasting/blob/master/Untitled.ipynb

We have converted this to Pytorch, and used other stock data:

https://colab.research.google.com/drive/1nChciJf6iMS36KGWQwzx82Mo9BABSm5_

Mamba model repository example we used as a reference for our Mamba model:

https://github.com/alxndrTL/mamba.py

Our Mamba model implementation for stock prediction:

https://colab.research.google.com/drive/1ts2MeYKUCTlVB9DmDoP4UWK-umUvxnZt

Our “Compare” program (we did not use a reference for this):

https://colab.research.google.com/drive/1IZnal3fnnJnFP1qSDsu8Jw75qnOuVBIC