Sophia Vasylyeva, Jennifer Galindo, Arath Dominguez

CS5393 Advanced Python Final Project

Stock Price Prediction Analysis Using LSTM and GRU Networks

Project Overview

This project implements a stock price prediction system using two different types of recurrent neural networks (RNNs) - Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks - to predict Apple Inc. (AAPL) stock prices. The implementation includes additional features such as sentiment analysis of stock-related news and confidence interval visualization to provide a more comprehensive analysis of the predictions.

Implementation Details

Data Processing and Model Architecture

The system fetches historical stock data using the Yahoo Finance API and preprocesses it using the MinMaxScaler to normalize the data between 0 and 1. The data is then structured into sequences with a 60-day window, which means each prediction is based on the previous 60 days of stock prices. This window size was chosen to capture both short-term trends and medium-term patterns in the stock movement.

Both LSTM and GRU models share a similar architecture:

- First layer: 50 units with return sequences enabled
- Dropout layer (0.2) for regularization
- Second layer: 50 units
- Another dropout layer (0.2)
- Dense output layer for final prediction

The models are trained with the Adam optimizer and mean squared error loss function, implementing early stopping and model checkpointing to prevent overfitting and save the best performing model.

Results Analysis

Model Performance

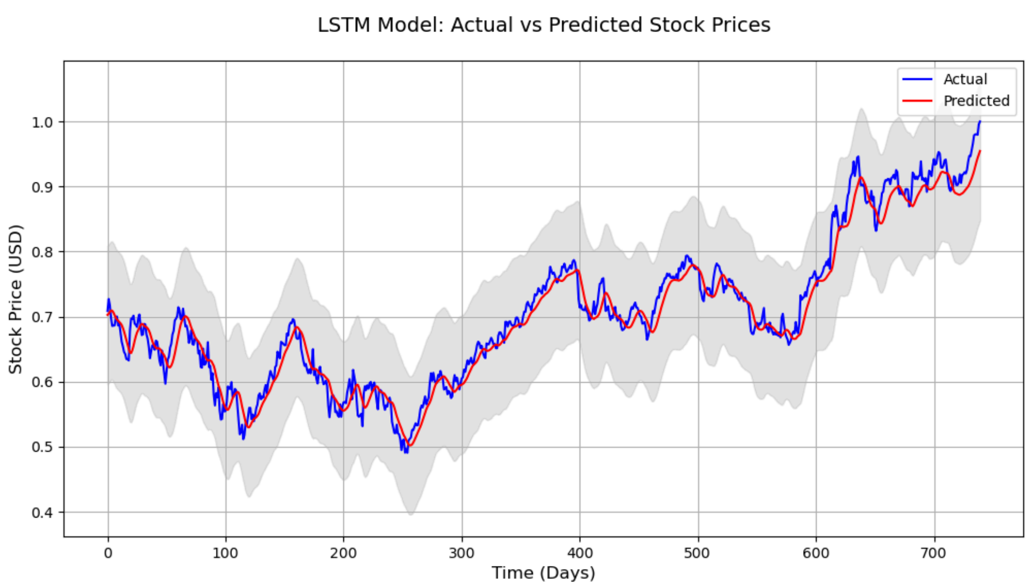
```
24/24 ━━━━━━━━━━━━━━━━━━━━ 0s 11ms/step
LSTM RMSE: {'rmse': 0.02277108367396496, 'mape': 17.614093159915658}
GRU RMSE: {'rmse': 0.015771431777037968, 'mape': 18.196940639681742}
24/24 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step
```

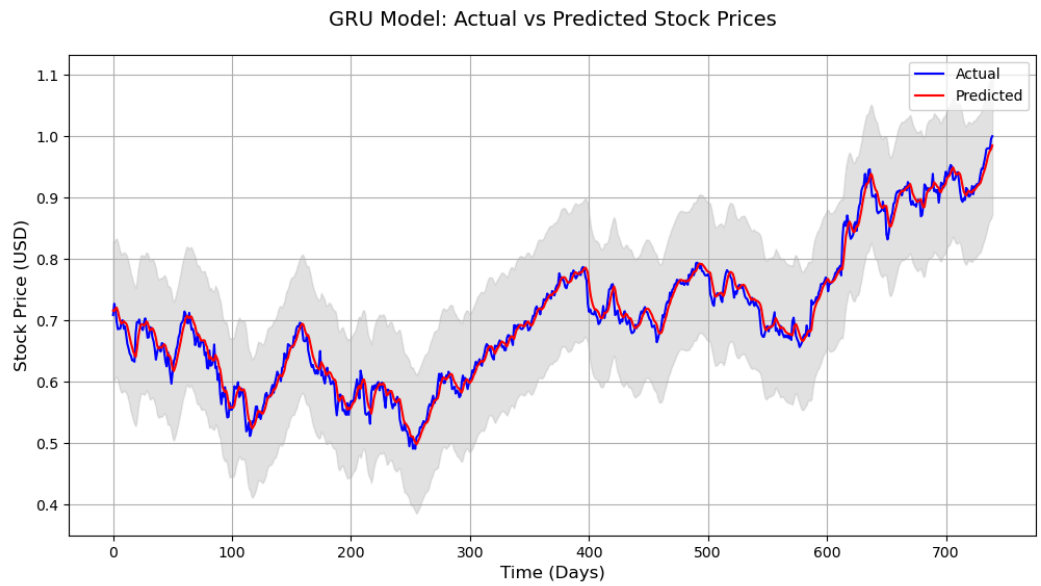
Figure 1: RMSE and MAPE Results

The GRU model achieved a slightly lower RMSE, suggesting better overall prediction accuracy, while the LSTM model showed a slightly better MAPE, indicating more consistent percentage-wise predictions. This difference in performance metrics suggests that each model might have different strengths in predicting various aspects of stock price movements.

Visualization and Confidence Intervals

The generated visualizations include an important feature: confidence intervals, represented by the gray shaded areas around the prediction lines. These intervals provide crucial information about the model's uncertainty in its predictions. The width of the shaded area represents one standard deviation above and below the predicted value, giving us a 68% confidence interval.





The confidence intervals notably widen in areas of higher volatility or uncertainty, which is particularly visible in the later portions of both graphs where the stock price shows more dramatic movements. This visual representation of uncertainty is valuable for risk assessment and decision-making, as it provides a range of likely outcomes rather than just a single prediction.

Sentiment Analysis Integration

The project incorporates sentiment analysis of stock-related news using NLTK's VADER sentiment analyzer. The test run with sample news articles about Apple produced a sentiment score of 0.4767, indicating a moderately positive sentiment. This integration of sentiment analysis provides an additional dimension to the prediction system, as market sentiment can be a leading indicator of stock price movements.

1. **Sentiment Analysis Scale**: The **SentimentIntensityAnalyzer** from the VADER sentiment analysis library assigns a compound score that ranges from -1 to +1:
   - **+1** indicates extremely positive sentiment.
   - **0** indicates a neutral sentiment.
   - **-1** indicates extremely negative sentiment.
2. **Interpretation of 0.4767**:
   - A score of **0.4767** is a moderately positive sentiment. It suggests that the majority of the articles or content analyzed have a positive tone.
   - This could reflect news such as praises about the company's performance, market performance, or positive investor sentiment. These findings can be seen in the visualizations as the stock price is currently at a peak.
3. **In Context**:

- If you're analyzing multiple news articles about a stock (for example, Apple), and you get a positive sentiment score, it could indicate that the articles are highlighting some positive factors (e.g., a market upturn, company achievements, etc.)
- A score closer to 0 would mean a neutral sentiment, where the articles aren't swaying significantly one way or another. A very negative sentiment score (e.g., -0.7 or -0.8) would indicate strong negative sentiment.

Challenges and Development Process

The development of this stock prediction system encountered several significant challenges that required careful resolution. The most pressing issue was determining optimal hyperparameters for both models. Initial attempts with 50 epochs led to overfitting, and early stopping with a patience of 5 was cutting off promising training runs prematurely. Through iterative testing, we settled on 20 epochs with a patience of 10, which provided the best balance between model performance and training efficiency. The GRU model consistently achieved better validation loss metrics, though it showed more sensitivity to sudden price changes.

The visualization component presented its own set of challenges, particularly in representing prediction uncertainty. The implementation of confidence intervals using standard deviation proved crucial, as initial attempts with fixed-width intervals failed to capture the varying levels of prediction uncertainty across different market conditions.

The sentiment analysis integration required careful consideration of score normalization and threshold selection for neutral sentiment filtering. Working with LLM guidance helped optimize these components, particularly in designing the sentiment score aggregation method and implementing the confidence intervals.

Interactions with LLM

The development process of the machine learning model was improved through interactions with LLMS. In terms of parameter optimization, the LLM helped fine-tune the model's hyperparameters, identifying optimal learning rates and batch sizes, while suggesting a dropout rate of 0.2 based on similar implementations.

The visualization component particularly benefited from LLM assistance, as it helped implement standard deviation-based confidence intervals and optimize the plotting code for better performance and readability. Initially, we plotted without confidence intervals.

The use of LLM provided valuable insights into architectural decisions, guiding the selection of layer sizes and types while explaining the complementary benefits of using both LSTM and

GRU models. The implementation of early stopping and model checkpointing came from this architectural guidance, significantly improving the model's training efficiency. Through this iterative collaboration with the LLM, we were able to successfully use the starter code and change it into a model more specific to the use case we developed during the planning phase.

This implementation demonstrates the potential of deep learning in stock price prediction while acknowledging the inherent uncertainties through confidence intervals and sentiment analysis integration. The combination of LSTM and GRU models provides complementary insights into price movements, while the visualization tools help in understanding the models' confidence in their predictions.

Code is shared in HTML format.