Title: Gold Price Prediction Using LSTM Neural Network: A Reproduction Study

Fatemeh Rajabi

Primary Paper: Varela, A. (2024). *Achilles, Neural Network to Predict the Gold Vs US Dollar Integration with Trading Bot for Automatic Trading.*

1 Introduction

Predicting the price of gold in the forex market is a challenging task due to the complex and dynamic nature of financial data. Gold prices fluctuate daily and are influenced by numerous economic factors such as inflation rates, currency movements, and geopolitical events. Making accurate short-term predictions could help investors make better-informed decisions. Since gold prices form a time series, using a model capable of handling sequential patterns is essential.

In this project, I reproduced parts of Angel Varela's work titled *"Achilles, Neural Network to Predict the Gold Vs US Dollar Integration with Trading Bot for Automatic Trading."* While the original project combined forecasting with an automated trading system, my focus was specifically on adapting and evaluating the forecasting component. Instead of minute-level data, I used daily data to develop an LSTM (Long Short-Term Memory) model that predicts the next day's gold price based on historical trends and technical indicators.

Many AI models developed in academic research are not tested in practical, real-world trading environments. Varela's Achilles project stands out because it used real market data and evaluated the model's performance in a realistic setting. By reproducing and adapting the forecasting part of his work, I aimed to better understand the strengths and limitations of using LSTM models for daily gold price prediction.

1.1 Open Questions

- Can LSTM-based models provide accurate predictions for daily gold price movements?
- How well does this approach work in practice, and could it be applied to real-world financial decision-making?

1.2 Brief Overview of the Approach

To address these challenges, I trained an LSTM model on historical gold price data, incorporating selected technical indicators as additional features. The model's performance was evaluated by comparing the predicted prices to actual market data, focusing on metrics like Mean Squared Error (MSE) and analyzing its potential applicability to real-world forecasting scenarios.

2 Background

Recent work by Angel Varela introduced a model called Achilles, a lightweight LSTM architecture designed to predict the price of gold versus the US dollar (XAUUSD). Varela trained the model on minute-level data and integrated it with a trading bot for automated decision-making, reporting a 162% profit over one month in a paper trading environment. While Varela's focus was on high-frequency trading, this study adapts his core forecasting model to daily data, aiming to evaluate its effectiveness in a more traditional time-series prediction setting without deploying an automated trading system.

The growing popularity of LSTM networks in financial forecasting is well supported in the literature. A systematic review by Ayitey Junior et al. analyzed 60 studies on forex market prediction and found that LSTM and ANN models are the most widely used algorithms, outperforming traditional techniques on metrics such as MAE, RMSE, and MAPE. Consistently, Siami-Namini et al. compared ARIMA and LSTM models for time-series forecasting and concluded that LSTM models are better suited for capturing nonlinear patterns — a frequent characteristic in financial data.

Beyond forex-specific research, Fischer and Krauss demonstrated that LSTMs outperform Random Forests and Logistic Regression when applied to S&P 500 daily stock data, showing the broader applicability of deep learning models across different financial markets. Complementing these studies, Yıldırım et al. investigated combining technical and macroeconomic indicators with LSTM networks, finding that integrating multiple sources of information further improves forecasting accuracy.

Other machine learning methods have also shown promise in financial prediction. Huang et al. demonstrated the effectiveness of Support Vector Machines (SVMs) in predicting stock market directional movement, highlighting their ability to handle nonlinear separations through kernel functions like RBF. However, they also emphasized the importance of robustness against overfitting, and LSTM models are increasingly recognized for sequential data tasks.

Further advancements include the work of Dasha et al., who introduced Bidirectional LSTM (Bi-LSTM) architectures that process input sequences both forward and backward, providing richer context at each timestep. Their results, along with an enhanced DeepSense Network optimized for forex forecasting, showed improved performance in predicting both price levels and directional trends compared to traditional and standard deep learning models.

Together, these studies reinforce the idea that LSTM-based architectures — especially when enhanced with hybrid features or bidirectional processing — are highly effective for financial forecasting tasks. This project builds upon that foundation by adapting and evaluating Varela's LSTM architecture for daily gold price prediction, aiming to make the model more interpretable and practical for broader market analysis beyond high-frequency trading.

3 Methods

3.1 Algorithm and Model Design

To predict daily gold prices, I developed a Long Short-Term Memory (LSTM) neural network, a specialized type of Recurrent Neural Network (RNN) designed to model sequential data and capture long-term dependencies. LSTM networks are particularly well-suited for financial forecasting because they can effectively learn temporal patterns over extended sequences, unlike traditional machine learning models that often assume data points are independent.

The forecasting model uses a sliding window approach: the LSTM network takes the past 60 days of data as input to predict the next day's closing price. To improve model performance and prevent overfitting, I employed two callbacks during training:

- **EarlyStopping**, which monitors validation loss and stops training if no improvement is observed.
- **ReduceLROnPlateau**, which reduces the learning rate when the validation performance plateaus.

The final model was evaluated using Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) on a separate test set to assess prediction accuracy.

3.2 Data Gathering and Preprocessing

3.2.1 Data Collection

I collected daily historical data for the GLD ETF, which tracks the gold price, using the yfinance library. The dataset included the following columns:

- **Open**: Price at market open
- **High**: Highest price of the day
- **Low**: Lowest price of the day
- **Close**: Price at market close
- **Volume**: Number of shares traded

The dataset spanned from January 1, 2015, to the present date.

3.2.2 Feature Engineering (Adding Technical Indicators)

To enhance the model’s understanding of market trends, I engineered additional features commonly used in technical analysis:

- **SMA_10**: 10-day Simple Moving Average of closing prices
- **SMA_30**: 30-day Simple Moving Average of closing prices
- **RSI_14**: 14-day Relative Strength Index, indicating overbought or oversold conditions

These indicators provide additional context about the momentum and smoothing of price trends.

3.2.3 Handling Missing Data

The calculation of moving averages and RSI introduced missing values at the start of the dataset. These incomplete rows were removed using the .dropna() function to ensure a clean dataset for training.

3.2.4 Normalization

Before feeding the data into the model, all features were scaled to a [0, 1] range using the MinMaxScaler from sklearn.preprocessing. Normalization ensures that features with larger numeric ranges (like volume) do not dominate the learning process.

4 Experiments

4.1 Baseline Model: ARIMA

To establish a baseline for comparison, I first implemented an ARIMA model with the configuration (p=3, d=2, q=2), selected based on tuning trials. The model was evaluated using Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) metrics. The results were:

- **ARIMA RMSE**: 16.69
- **ARIMA MAE**: 13.91

While ARIMA was able to capture the overall upward trend in gold prices, it struggled with modeling the true fluctuations and turning points. This made it less useful for practical short-term trading, where predicting local volatility is critical.

4.2 Main Model: LSTM

For the main predictive model, I built a Long Short-Term Memory (LSTM) neural network trained on historical gold price data enhanced with technical indicators. The model input consisted of sequences representing the past 60 days, and the task was to predict the next day's closing price.

4.2.1 Model Architecture:

- Two LSTM layers with 100 units each
- Two dropout layers (rate = 0.2) to prevent overfitting
- One dense output layer
- Compiled using the Adam optimizer and Mean Squared Error (MSE) loss function

4.2.2 Training Setup:

- 50 epochs
- Batch size of 32
- Callbacks:
  - EarlyStopping (patience=5)
  - ReduceLROnPlateau (patience=3, factor=0.5)
- Data split: 80% training, 20% testing (without shuffling to preserve temporal order)

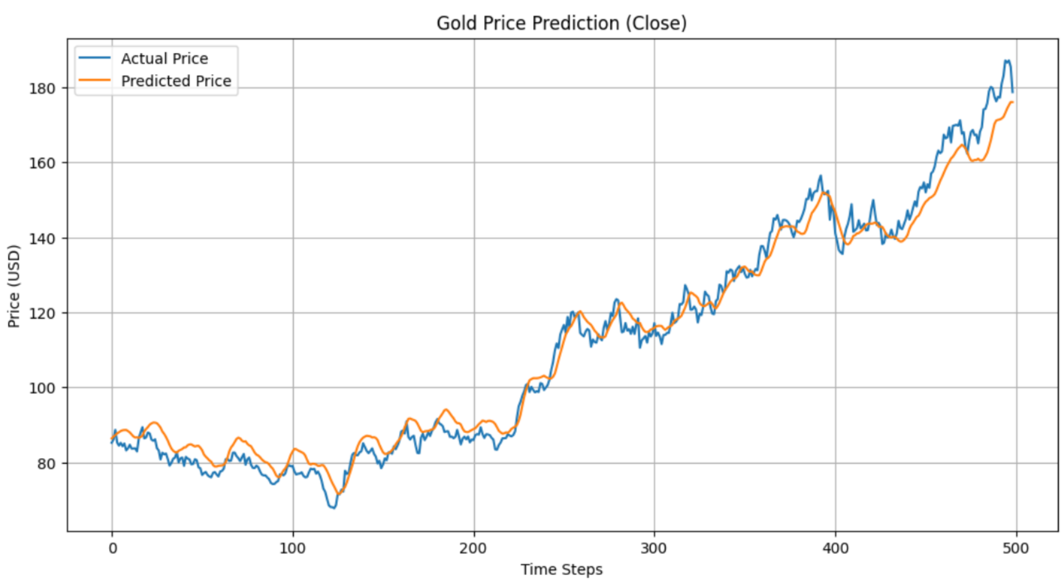
4.3 Results and Observations

The LSTM model outperformed ARIMA by a wide margin, achieving significantly lower RMSE and MAE scores. As shown in the figure below, the LSTM predictions closely tracked the actual closing prices, capturing not only the general trend but also most of the peaks and valleys.

While minor deviations between actual and predicted values existed over longer horizons, the LSTM model demonstrated strong potential for daily forecasting tasks. The use of technical indicators like moving averages (SMA) and RSI further improved model accuracy by providing additional market context.

After testing various model configurations and tuning hyperparameters, the best-performing LSTM model achieved:

- Root Mean Squared Error (RMSE): 4.37
- Mean Absolute Error (MAE): 3.54



4.5 Reproduction Analysis

Although I adapted Varela's approach by focusing on daily instead of minute-level predictions, the results aligned with the core finding that LSTM models are highly effective for financial time-series forecasting. However, because I did not implement the full trading system or high-frequency prediction as in the original paper, a direct one-to-one reproduction was not possible. My experiments confirm the predictive strength of LSTM models but in a different timescale and context, focusing on daily market dynamics rather than high-frequency trading.

This study did not extend to testing with different datasets beyond the GLD ETF. However, based on the strong performance observed here, future work could involve testing the model on other commodities or currencies to evaluate its generalization capability across different financial assets.

5 Conclusions

LSTM is an effective model for daily gold price forecasting, thanks to its ability to learn sequential patterns. It clearly outperformed the baseline ARIMA model, with lower error rates and better trend tracking. Although I did not include a trading bot like in the original Achilles paper, this study shows that LSTM alone can provide strong predictive performance on time series financial data.

Although the LSTM model outperformed ARIMA, its predictions still showed a lag near turning points. Additionally, the model's performance may be sensitive to hyperparameters and external economic shocks not captured in the dataset.

Through this project, I learned how to apply deep learning techniques for financial time series forecasting, especially how to manage sequential dependencies and prevent model overfitting. One of the main challenges was tuning the LSTM model to balance bias and variance, which I addressed by using EarlyStopping and learning rate adjustments through ReduceLROnPlateau.

In future work, I plan to:

- Explore sentiment analysis from financial news
- Use more advanced models like Bidirectional LSTM or Transformers

References

1. Varela, A. (2024). *Achilles, Neural Network to Predict the Gold Vs US Dollar Integration with Trading Bot for Automatic Trading.* https://arxiv.org/abs/2410.21291
2. Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research, 270*(2), 654–669. https://doi.org/10.1016/j.ejor.2017.11.054
3. Siami-Namini, S., Tavakoli, N., & Siami Namin, A. (2018). A comparison of ARIMA and LSTM in forecasting time series. *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. https://doi.org/10.1109/ICMLA.2018.00227
4. Ayitey Junior, M., Appiahene, P., Appiah, O., & Bombie, C. N. (2023). Forex market forecasting using machine learning: Systematic literature review and meta-analysis. *Journal of Big Data, 10*(9). https://doi.org/10.1186/s40537-022-00676-2
5. Nguyen, P. D., Thao, N. N., Chi, D. T. K., Nguyen, H. C., Mach, B. N., & Nguyen, T. Q. (2024). Deep learning-based predictive models for forex market trends: Practical implementation and performance evaluation. *Science Progress, 107*(3), 1–27. https://doi.org/10.1177/00368504241275370
6. Huang, W., Nakamori, Y., & Wang, S.-Y. (2005). Forecasting stock market movement direction with support vector machine. *Computers & Operations Research, 32*(10), 2513–2522. https://doi.org/10.1016/j.cor.2004.03.016
7. Yıldırım, D. C., Toroslu, I. H., & Fiore, U. (2021). Forecasting directional movement of Forex data using LSTM with technical and macroeconomic indicators. *Financial Innovation, 7*(1). https://doi.org/10.1186/s40854-020-00220-2
8. Dasha, S., Sahu, P. K., & Mishra, D. (2023). Forex market directional trends forecasting with Bidirectional-LSTM and enhanced DeepSense network using all member-based optimizer. *Intelligent Decision Technologies, 17*(3), 1351–1382. https://doi.org/10.3233/IDT-230183

Sharing Agreement

I agree to share my work as an example for next semester without hiding my name.