Predicting Stock Prices with Long Short-Term Memory Networks in TensorFlow

By: Donovan Johnson

Abstract

This paper explores the application of Long Short-Term Memory (LSTM) networks for stock price prediction using TensorFlow. It presents a machine learning model designed to capture temporal dependencies in historical stock data and forecast future prices. The paper details the model architecture, training process, and evaluation metrics. The results demonstrate the model's ability to learn patterns and predict stock prices with a certain degree of accuracy.

Introduction

Stock price prediction is a captivating and challenging task in the financial domain. Traditional methods often rely on statistical analysis or technical indicators, which might not effectively capture the complex dynamics of the stock market. Machine learning approaches, particularly recurrent neural networks (RNNs), have emerged as promising alternatives due to their ability to learn from historical data and identify underlying patterns.

LSTMs, a specific type of RNN, excel at handling sequential data like stock prices. Their internal architecture allows them to process past information and make informed predictions about the future. This paper implements an LSTM model using TensorFlow, a popular deep learning framework, to forecast stock prices.

Methods

Data Collection:

Historical stock price data for Apple Inc. (AAPL) from January 1, 2010, to December 31, 2023, are obtained from Yahoo Finance using the yfinance Python library.

Data Preprocessing:

The closing prices of AAPL stock are selected from the retrieved dataset and normalized using MinMaxScaler to scale the values between 0 and 1.

Model Architecture:

An LSTM neural network is constructed using the TensorFlow and Keras libraries. The model architecture consists of a single LSTM layer with 50 units, followed by a dropout layer with a dropout rate of 0.2 to prevent overfitting. The output layer consists of a single neuron, predicting the next day's closing price.

Training and Testing:

The dataset is split into training and testing sets, with 80% of the data used for training and the remaining 20% for testing. A look-back window of 50 days is chosen for creating the input sequences for the LSTM model.

Model Training:

The LSTM model is trained using the training data with a mean squared error loss function and the Adam optimizer for 100 epochs.

Prediction and Forecasting:

The trained model is used to predict the stock prices for the testing dataset. Additionally, the model is employed to forecast the next six months of AAPL stock prices.

Discussion

The results demonstrate that the LSTM model can learn from historical data and predict stock prices with a certain level of accuracy. However, it's crucial to acknowledge the inherent limitations:

- **Market Volatility:** The stock market is inherently volatile and influenced by various unpredictable factors. The model's predictions should not be misconstrued as absolute certainties.
- **Data Quality:** The model's performance heavily relies on the quality and quantity of training data. Insufficient or unreliable data can lead to inaccurate predictions.
- **Hyperparameter Tuning:** The model's performance can be improved through further hyperparameter tuning, potentially involving techniques like grid search or random search.

Conclusion

This paper presented an LSTM model built using TensorFlow to predict stock prices. The model demonstrates the potential of machine learning for stock price forecasting. Future work can explore incorporating additional features, ensemble methods with other prediction models, and more advanced LSTM architectures for potentially better prediction accuracy.