Cryptocurrency Price Prediction Using LSTM Neural Networks with TensorFlow.js

Fatwa Kurnia Budiman
Independent Researcher

Abstract

Cryptocurrency price prediction is a challenging task due to the high volatility and complex market dynamics. This paper presents a machine learning approach using Long Short-Term Memory (LSTM) networks implemented with TensorFlow.js to predict future prices of cryptocurrencies based on historical price and volume data. The model is trained on a sequence of normalized data points and uses a sliding window of 30 days to forecast prices for the next days. Model training and inference are performed client-side with the capability to save and load models from a backend server. Experimental results show promising accuracy in short-term price forecasting, demonstrating the feasibility of browser-based deep learning applications for financial predictions.

Keywords

Cryptocurrency, Price Prediction, LSTM, TensorFlow.js, Deep Learning, Time Series Forecasting

I. Introduction

Cryptocurrencies have gained widespread attention as alternative investment assets characterized by significant price volatility. Accurate price prediction can help investors make informed decisions, but traditional statistical models often fail to capture complex temporal dependencies. Recurrent neural networks (RNN), especially Long Short-Term Memory (LSTM) networks, have been shown to effectively model sequential data and temporal patterns in financial markets.
This work implements a cryptocurrency price prediction system using LSTM networks directly in the browser with TensorFlow.js. The model leverages historical price and volume data retrieved via the CoinGecko API, trained on sequences of 30 days to predict subsequent prices. The system supports saving trained models on a backend server and reloading them to avoid retraining from scratch, enabling incremental learning and multi-user scalability.

II. Related Work

Several studies have explored cryptocurrency price prediction using machine learning techniques. Prior work includes ARIMA models, support vector machines, and various deep learning architectures such as LSTM and gated recurrent units (GRU). While many implementations require backend servers or cloud GPUs, our approach utilizes TensorFlow.js to perform training and inference entirely on the client-side, reducing latency and increasing accessibility.

III. Methodology

A. Data Collection
Price and volume data for cryptocurrencies are obtained using the CoinGecko API, focusing on the top 100 coins by trading volume. Data include daily historical prices and total trading volumes for a configurable number of past days.

B. Data Preprocessing
Raw prices and volumes are normalized using min-max scaling to map values between 0 and 1. Data sequences are created using a sliding window of 30 days (timesteps), each consisting of normalized price and volume pairs as features, with the next day's normalized price as the target label.

C. Model Architecture
The LSTM model is constructed as a sequential TensorFlow.js model consisting of:
- One LSTM layer with 50 units, processing sequences of shape (30, 2).
- One Dense output layer with a single neuron for regression output.

The model is compiled using the Adam optimizer and mean squared error loss function.

D. Training and Prediction
The model is trained for 50 epochs on the prepared dataset. After training, the model can predict prices iteratively for the desired number of future days by feeding the last 30-day sequence and sliding it forward with predicted values.

E. Model Persistence
Trained models are saved to a backend PHP server using multipart form uploads, including model topology (JSON) and weights (binary). Models can be loaded back to resume predictions without retraining.

IV. Experimental Results

The system was tested on various cryptocurrencies, with training and prediction running entirely in-browser. Training progress is visualized with a progress bar and rotating training hints for user engagement. The model achieved reasonable predictive performance in short-term forecasting, although accuracy depends heavily on data quality and length. Historical price charts and prediction results are displayed dynamically, with normalized data denormalized for user-friendly output.

V. Conclusion

This paper demonstrates the feasibility of deploying deep learning models for cryptocurrency price prediction directly in web browsers using TensorFlow.js. The LSTM-based approach, coupled with server-side model persistence, allows scalable multi-user scenarios without heavy backend dependencies. Future work may involve integrating sentiment analysis, additional features, and improving model robustness.

References

[1] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[2] C. Aggarwal, *Neural Networks and Deep Learning: A Textbook*, Springer, 2018.

[3] T. Brown, et al., "TensorFlow.js: Machine Learning for the Web and Beyond," *Proc. 3rd SysML Conf.*, 2019.

[4] CoinGecko API Documentation. [Online]. Available: https://www.coingecko.com/en/api