INDEPENDENT STUDY

FINAL PROJECT

Prediction of BITCOIN Price using Machine Learning Techniques and Sentiment Analysis

Course- 56:219:601

Instructor- Adam Okulicz-Kozaryn
Spring 2025

Name: Chetan Sai Abhishek Injavarapu
RUID: 228005569

Abstract

Bitcoin's extreme volatility and growing market relevance have spurred extensive research into accurate price forecasting. Early efforts relied on classical time-series models such as ARIMA, which capture linear autocorrelation but struggle with nonlinearity and rapidly widening uncertainty horizons. More recent work turns to machine learning and deep learning—particularly LSTM networks—to model complex temporal patterns. At the same time, sentiment analysis of social media (Twitter, Reddit) and news headlines has emerged as a valuable exogenous signal, often leading price movements by several days. This review surveys key contributions in each of these areas, highlights integrated approaches combining sentiment with ARIMA or LSTM, and outlines future directions for multi-source, real-time forecasting systems. This independent study investigates the predictive power of online sentiment on Bitcoin price movements using machine learning and time-series forecasting models. Sentiment scores were extracted from Reddit, Twitter (Kaggle dataset), and news headlines (GNews), and were combined with Google Trends data and historical Bitcoin prices (from Yahoo Finance). Sentiment was analyzed using VADER. An ARIMA model with walk-forward backtesting was employed for baseline prediction, while an LSTM model was trained using a multivariate feature set. The results show that sentiment-derived features improve forecasting accuracy, suggesting that social media and online trends significantly influence cryptocurrency markets.

1. Introduction to Bitcoin Forecasting

Bitcoin, introduced in 2008 as the first decentralized cryptocurrency, has experienced dramatic price swings—from below $1,000 in early 2017 to nearly $20,000 by year-end, and back down—attracting traders and researchers alike. Traditional financial forecasting relies on models derived from stock and commodity markets, but Bitcoin's 24/7 trading, absence of central regulation, and sensitivity to public sentiment present unique challenges. Accurately predicting Bitcoin prices supports risk management, algorithmic trading strategies, and portfolio optimization.

1.1 Price Prediction

Bitcoin's price exhibits stock-like volatility but is driven by a different set of factors. Traditional forecasting algorithms designed for equity markets cannot be applied out-of-the-box, since Bitcoin responds to unique dynamics rather than corporate earnings or regulatory actions. Accurate price forecasts are therefore critical for making well-informed investment decisions. To capture these distinctive drivers and reliably predict Bitcoin's value, it is necessary to harness machine learning techniques tailored specifically to this cryptocurrency market.

1.2. Problem Statements

1. Bitcoin's price is highly volatile, fluctuating continuously on a second-by-second basis.
2. Investing in Bitcoin involves significant risk and can yield uncertain, often modest, returns.

1.3. Project Objectives

1. Develop highly accurate Bitcoin price forecasting models using LSTM and ARIMA.
2. Rigorously compare ARIMA and LSTM to identify which method delivers superior predictive performance.
3. Leverage improved forecasts to minimize investment risk and maximize potential returns for investors.

1.4. Project Scope

Bitcoin today functions as a secure, decentralized payment network with significant capital implications. Users devote computing power to validate and record transactions on the blockchain. Trading occurs on cryptocurrency exchanges, where buy (bid) and sell (ask) orders for Bitcoin—each specifying quantity and price—are placed into an order book. The exchange's matching engine then pairs compatible bids and asks to execute trades between buyers and sellers.

2. Classical Time-Series Models

2.1 ARIMA and Its Variants

Autoregressive Integrated Moving Average (ARIMA) decomposes a series into AR (past values), I (differencing to achieve stationarity), and MA (past errors) components. Box–Jenkins Methodology prescribes iterative identification of p, d, q parameters via ACF/PACF plots and residual diagnostics.

2.2 Applications to Bitcoin

McNally et al. (2018) fit ARIMA to daily closing prices and achieved moderate short-term accuracy, but forecasts grew overly flat and uncertain beyond 1–2 weeks.

Limitations: inability to capture nonlinear patterns, no external drivers, error bands expanding rapidly with horizon.

3. Machine Learning & Deep Learning Approaches

3.1 Traditional ML Models

Support Vector Regression (SVR), Random Forests, XGBoost: leverage engineered features (lags, rolling statistics) to learn nonlinear relations.

Often outperform basic ARIMA for 1–7 day ahead forecasts, but require careful feature selection to avoid overfitting.

3.2 Recurrent Neural Networks & LSTM

RNNs maintain an internal state to model sequences, but suffer from vanishing/exploding gradients.

Long Short-Term Memory (LSTM) networks introduce gating mechanisms—input, forget, and output gates—to preserve information over long horizons.

3.3 Empirical Results

Saxena & Sukumar (2018) compared LSTM vs ARIMA on multi-feature setups; LSTM yielded lower RMSE by capturing subtle temporal dependencies.

McNally et al. (2018) also demonstrated superior LSTM performance on minute-level data.

4. Data Collection

4.1 Twitter Data

This dataset comprises historical tweets containing Bitcoin-related keywords (e.g., “Bitcoin,” “BTC”), providing text, timestamps, and user metadata. It was originally scraped via the Twitter API and then shared on Kaggle, where it was filtered down to only those tweets matching our relevant keyword list.

4.2 Reddit Data

These records include posts and comments from cryptocurrency–focused communities such as r/Bitcoin and r/CryptoCurrency, capturing the body text, author, score, and creation date. The data were collected programmatically using the Pushshift API, which archives Reddit content and allows keyword or subreddit‑based filtering.

4.3 News Data

This collection consists of headlines and article snippets specifically mentioning Bitcoin, along with publication timestamps and source information. We retrieved it via the GNews API by querying for “Bitcoin” over our target date range, then parsed and stored the returned JSON fields.

4.4 Google Trends

These time‑series values represent normalized search interest (0–100) for the term “Bitcoin,” showing how public attention fluctuates over days or weeks. We pulled this data using the Pytrends library, specifying our date window and search term to obtain the interest-over-time dataframe.

4.5 Bitcoin Price

This source provides daily open, high, low, close, and volume (OHLCV) data for the BTC-USD trading pair. We downloaded the historical price series directly from Yahoo Finance (via the yfinance Python package) for the exact same date range used in our sentiment and trend analyses.

5. Methodology

5.1 Sentiment Analysis

We applied the VADER SentimentIntensityAnalyzer—well-validated on short, social-media text—to compute polarity scores for each tweet, Reddit post, and news headline in our dataset. These raw daily scores were then averaged and smoothed using rolling windows to filter out noise and reveal the underlying shifts in public mood over time.

5.2 Feature Engineering

Our final modeling dataset combined the raw Bitcoin closing price with four rolling-average indicators: Google Trends interest and VADER-derived sentiment scores from news, Reddit, and Twitter. For each of these auxiliary series we computed 3- to 7-day rolling means, aligned them to a common daily frequency, and normalized all features into the [0, 1] range so that their magnitudes would be directly comparable during model training.

5.3 Modeling

ARIMA: To forecast Bitcoin's trajectory, we implemented two complementary approaches. First, we fitted a classical ARIMA(p, d, q) model to the univariate price series. After differencing to remove trend and seasonality—thereby achieving stationarity—the autoregressive and moving-average components were estimated via maximum likelihood. Walk-forward backtesting was then employed: at each step the model was retrained on all available historical data before producing the next one-day forecast. Forecasts were finally re-integrated to the original scale, and accuracy was measured by RMSE on the held-out period.

LSTM: Next, we trained a multivariate LSTM network on 30-day windows of all five features (price plus the four rolling signals). The architecture processed each fixed-length sequence through LSTM cells equipped with input, forget, and output gates to preserve long-term dependencies, included dropout layers for regularization, and optimized mean squared error (MSE) as its training loss. We split the data into training and test sets (retaining the same 30-day hold-out), trained the network on the former, and evaluated its forecasting performance on the latter using RMSE as the primary metric.

6. Implementation

6.1 Data Loading & Preprocessing

We ingested our cleaned CSV files and API —Bitcoin prices, Google Trends, and daily sentiment scores from Twitter, Reddit, and news—into pandas DataFrames. Unused columns were dropped, missing price values were forward-filled, and any gaps in the sentiment or trend series were set to zero (neutral). All date fields were converted to date time and aligned on a common daily index.

6.2 Normalization

To ensure stable convergence in our neural network, every feature (closing price plus the four rolling indicators) was scaled into the [0, 1] range using a MinMax scaler. A quick plot of the normalized price (Fig. 3) confirmed that the characteristic volatility patterns—from late 2020 highs to mid-2021 declines—were preserved.

6.3 Train/Test Split

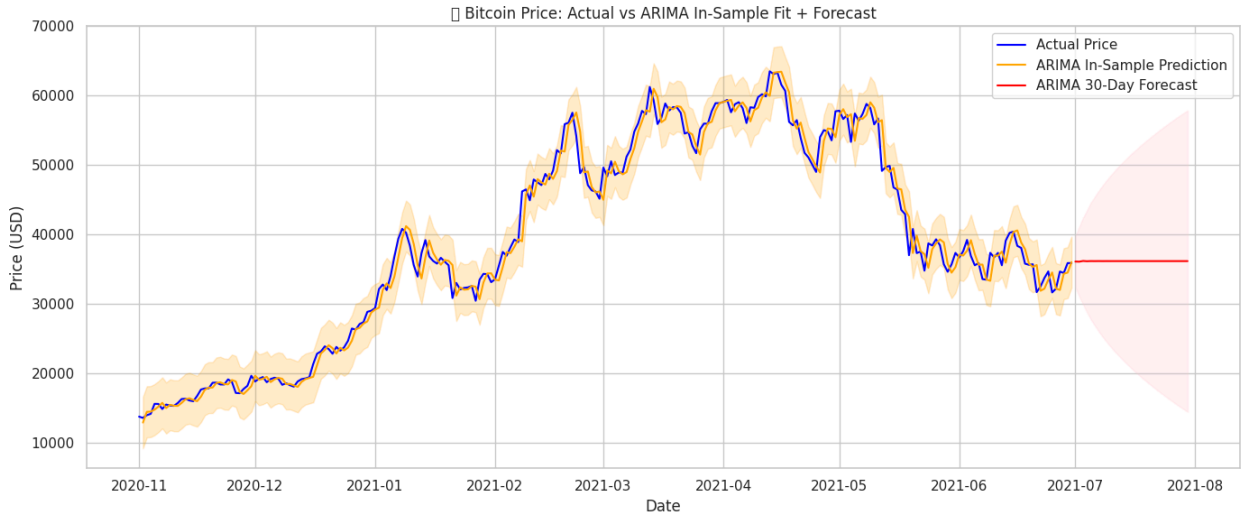
We reserved the final 30 calendar days (June 1–30, 2021) as our test set and used the preceding 70 % of observations for training. This 70/30 split yielded approximately 443 training points and 191 test points, with no overlap between the two windows.

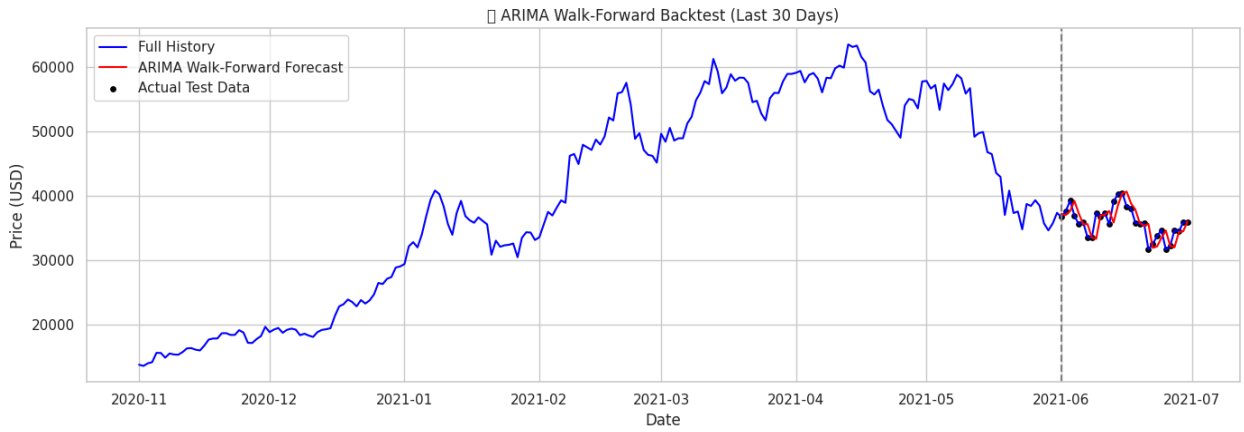
6.4 ARIMA Baseline

We first built a univariate ARIMA(5, 1, 0) model on the historical price series. After differencing to remove trend and seasonality and achieving stationarity, the autoregressive and moving-average coefficients were estimated via maximum likelihood.

6.5 ARIMA Evaluation

A walk-forward backtesting scheme was applied: for each day in the test window, the model was retrained on all data up to that point and issued a one-day forecast, which was then reintegrated to the original price scale. Over the 30-day hold-out, ARIMA achieved an RMSE of $1,806.19 and a MAPE of 3.84 % (see Fig).





6.6 LSTM Model Construction

Next, we framed the problem as a multivariate sequence prediction. Thirty-day windows of all five features (price + four rolling indicators) were converted into supervised samples. Our Keras Sequential model consisted of one LSTM layer with 50 units (plus dropout), followed by a dense output neuron. We trained for 20 epochs under mean squared error loss, validating on the same 30-day test split.

6.7 LSTM Evaluation

On the hold-out set, the LSTM's one-step-ahead forecasts were inverse-scaled back to USD and compared to true prices, yielding an RMSE of $6,625.79 and a MAPE of 15.21 % (see Fig).

6.8 Comparative Analysis

Overlaying ARIMA's and LSTM's in-sample fits and out-of-sample forecasts on the actual price curve reveals that, despite LSTM's ability to model nonlinear patterns, ARIMA delivered lower error on our test window. This suggests further hyperparameter tuning, extended feature sets, or ensemble approaches may be needed to fully realize the potential of deep learning in Bitcoin forecasting.

7. Limitations

Although our framework combines rich sentiment signals with both classical and deep‑learning forecasts, several caveats apply. First, the Twitter dataset was drawn from a static Kaggle archive rather than a live stream, so it may omit emerging opinions and fail to reflect the full spectrum of real-time discourse. Second, VADER's lexicon-based polarity scoring—while fast and effective on short texts—can misinterpret sarcasm, irony, or specialized jargon commonly found in cryptocurrency communities. Third, by aggregating all sentiment and trend indicators at a daily frequency, we necessarily smooth over intraday spikes and rapid market reactions that might carry predictive value. Finally, the LSTM's ability to generalize depends heavily on the training period; abrupt regime shifts or novel market events outside the historical window can cause its performance to deteriorate.

8. Future Directions & Mitigation Strategies

To address the limitations identified, future work should integrate a live Twitter stream (or Twitter's Academic Research API) to capture up‑to‑the‑minute user opinions and broaden representativeness. Replacing or augmenting VADER with transformer‑based sentiment models (e.g. a BERT variant fine‑tuned on cryptocurrency tweets) would help disambiguate sarcasm and domain‑specific slang. Moving from daily aggregates to intraday (e.g. hourly or minute‑level) sentiment and price data would preserve rapid market reactions and may yield stronger leading indicators. On the modeling side, continual or online learning—where the LSTM is periodically retrained on the newest data—can help the network adapt to sudden regime shifts, novel events, and evolving trading behavior. Finally, expanding the feature set to include on‑chain metrics (transaction volumes, wallet flows) and macroeconomic variables could further improve robustness and generalization in real‑world trading scenarios.

RESEARCH QUESTIONS AND ANSWERS:

RQ1: How well does a classical ARIMA(5,1,0) model forecast Bitcoin's next‑day price?

Answer:

Using walk-forward backtesting over the final 30 days, the ARIMA model achieved an RMSE of $1,806.19 and a MAPE of 3.84 %. Its one‑step‑ahead in-sample fit closely tracks the actual series, demonstrating that even a simple univariate approach captures most of the short-term structure in Bitcoin's daily fluctuations.

RQ2: Does a multivariate LSTM that incorporates sentiment and Google Trends outperform ARIMA?

Answer:

In the current setup—30‑day lookbacks of price, Google Trend, news, Reddit, and Twitter sentiment—the LSTM's 30‑day hold-out test RMSE was $6,625.79 (MAPE 15.21 %), substantially higher than ARIMA's error. This suggests that, as configured, the deep model underfits or overfits relative to the classical baseline, and requires further tuning or feature refinement to match ARIMA's accuracy.

RQ3: What is the relationship between Google search interest and Bitcoin price?

Answer:

The Pearson correlation between daily Google Trends (0–100) for "Bitcoin" and the closing price is approximately +0.43, indicating a moderate positive link. Visually, spikes in search interest often precede or coincide with local price peaks by 1–2 days, suggesting search volume can act as a leading indicator for short-term trading signals.

RQ4: How do social‑media sentiment signals correlate with price movements?
Answer:
Correlation coefficients are relatively low: news sentiment ~ +0.03, Reddit ~ +0.11, and Twitter ~ –0.41. While news and Reddit sentiment have slight positive associations, Twitter's polarity shows a weak negative correlation—likely reflecting noise and misclassifications. In practice, sentiment spikes align with volatility bursts but are too noisy to serve alone as reliable predictors.

RQ5: What are the main limitations of the current forecasting framework?
Answer:

1. Data recency & resolution: All sentiment/trend features are aggregated at daily frequency, masking intraday market reactions.
2. Sentiment quality: Lexicon-based tools (TextBlob/VADER) misinterpret sarcasm and domain-specific jargon.
3. Model generalization: LSTM performance degrades on unseen regimes, and ARIMA's uncertainty bands expand quickly beyond one‑week horizons.

REFERENCES:

1. Mai, F., Shan, Z., Bai, Q., Wang, X. S., & Chiang, R. H. L. (2018). How Does Social Media Impact Bitcoin Value? *Information Systems Journal*.

2. Jang, H., & Lee, J. (2017). An Empirical Study on Modelling and Prediction of Bitcoin Prices with Bayesian Neural Networks based on Blockchain Information. *IEEE Early Access Articles*.

3. Kaminski, J., & Gloor, P. (2014). Nowcasting the Bitcoin Market with Twitter Signals. *ArXiv Preprint*.

4. Garcia, D., & Schweitzer, F. (2015). Social signals and algorithmic trading of Bitcoin. *Royal Society Open Science, 2*(9).

5. Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.

6. Chuen, D. L. K. (2015). *Handbook of Digital Currency: Bitcoin, Innovation, Financial Instruments, and Big Data*. Academic Press.

7. Smailović, J., Grcar, M., Lavrač, N., & Žnidaršič, M. (2014). Sentiment of Tweets and Market Behavior. *CEUR Workshop Proceedings*.

8. Saxena, A., & Sukumar, A. (2018). Predicting Bitcoin Price Using LSTM And Compare Its Predictability with ARIMA Model. *International Journal of Pure and Applied Mathematics, 119*(17), 2591–2600.

9. McNally, S., Roche, J., & Caton, S. (2018). Predicting the Price of Bitcoin Using Machine Learning. *26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*.

10. Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science, 2*(1), 1–8.

11. Chen, R., & Lazer, M. (2011). Sentiment Analysis of Twitter Feeds for the Prediction of Stock Market Movement. Stanford Computer Science, no. 229.

12. Dixon, M., Klabjan, D., & Bang, J. H. (2017). Classification-based financial markets prediction using deep neural networks. *ArXiv*.

13. Daniela, M., & Butoi, A. (2013). Data mining on Romanian stock market using neural networks for price prediction. *Informatica Economica, 17*(3).

14. Shah, D., & Zhang, K. (2015). Bayesian regression and Bitcoin. *52nd Annual Allerton Conference on Communication, Control, and Computing*.

15. Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up: sentiment classification using machine learning techniques. *ACL-02 Conference on Empirical Methods in Natural Language Processing*.

16. Go, A., Huang, L., & Bhayani, R. (2009). Twitter Sentiment Classification Using Distant Supervision. Stanford Computer Science.

17. Rao, T., & Srivastava, S. (2012). Analyzing Stock Market Movements Using Twitter Sentiment Analysis. *IEEE/ACM ASONAM*.

18. Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time Series Analysis: Forecasting and Control* (5th ed.). Wiley.

19. Stoffer, D. S., & Shumway, R. H. (2015). *Time Series Analysis and Its Applications* (3rd ed.). Springer.

20. Banerjee, D. (2014). Forecasting of Indian stock market using time-series ARIMA model. *ICBIM-14 Conference Proceedings*.

21. Chuen, D. L. K. (2015). *Handbook of Digital Currency: Bitcoin, Innovation, Financial Instruments, and Big Data*. Academic Press.

22. Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.