

Time Series Analysis for Bitcoin Price Prediction using Prophet

Deepu John (20BDS0373), Rohit Ketan Deval (20BKT0103), Aryan Khare (20MIC0103), Aditya Sarangarajan (20BCE0985)

INTRODUCTION

1.1 Overview

Time series analysis is a statistical method that uses historical data to predict future values. It can be used to predict a wide variety of time series data, including cryptocurrency price - such as the price of Bitcoin. In this project, we use Prophet – which is a Python library that can be used for time series analysis.

1.2 Purpose

The purpose of this project is to perform time series analysis for Bitcoin price prediction using Prophet and to forecast future price movements and trends in the Bitcoin market based on historical price data.

2 LITERATURE SURVEY

Based on a previous study using LSTM, McNally et al. [9] proposed a Bitcoin price method using recurrent neural networks (RNN) and combined RNN with LSTM [10] to propose an automated cryptocurrency price prediction using machine learning techniques based on historical trends (daily trends). Pant et al. [11] proposed a prediction of Bitcoin price with looping artificial neural network techniques based on Twitter sentiment, the results of which are quite impressive, showing the relationship between sentiment and LSTM results [12]. Wu et al. [12] developed a new framework for predicting the price of bitcoin using LSTM and suggested two different LSTM models: standard LSTM and LSTM with autoregressive integrated moving average (ARIMA) with 208 record datasets, compared to mean absolute error (MSE), root mean square error (RMSE), and mean absolute percentage error (MAPE).

Existing problem

Traditional time series forecasting models face the following issues:

1. Time series forecasting models like ARIMA often require manual identification and specification of seasonal components, which can be challenging and time-consuming.
2. Traditional models struggle to adapt to trend changes, leading to less accurate forecasts. Prophet's trend flexibility allows it to better capture and adapt to these variations in the data.

3. Most models fail to handle missing data and outliers which are common in real-world time series datasets and can significantly impact forecast accuracy.

2.2 Proposed solution

1. Data Collection and Preprocessing: We download Bitcoin price data using the Yahoo Finance API. We used 2 years daily data between 2018 and 2019. We then performed data preprocessing by removing unnecessary columns and renaming column names.

2. Split data: We then split the preprocessed dataset into training and test sets. We choose a cutoff data and the data before the cutoff date is the training dataset, and the data after the cutoff date is used as the testing dataset. '2019-11-30' was used as the cutoff date. The first 23 months are used for model training, and the last month is used for testing.

3. Model training: We then trained the time series model using the training dataset.

``interval_width`` specifies the prediction interval. We changed the default value of 80% to 95% prediction interval. It makes the upper bound and the lower bound of the prediction broader.

``n_changepoints`` is the number of change points in the time series trend. The default value is 25. Based on the shape of the Bitcoin price data, it was set at 7.

Model prediction: We then used the trained Prophet model to make the prediction. We use the last 31 days to create the future dataframe. This is the same as using the testing dataset we created above. The prediction output contains lots of information. We kept the predicted value ``yhat`` and its prediction interval upper and lower bound value.

4. Cross Validation: Then we did cross-validation for the time series model. Prophet has a ``cross_validation`` function to automate the comparison between the actual and the predicted values.

5. Model Evaluation: Finally, we assessed the model's performance using appropriate evaluation metrics, such as: MSE (Mean Squared Error), RMSE (Root Mean Square Error), MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error), MDAPE (Median Absolute Percentage Error) and SMAPE (Symmetric Mean Absolute Percentage Error)

3 THEORITICAL ANALYSIS

3.1 Hardware / Software

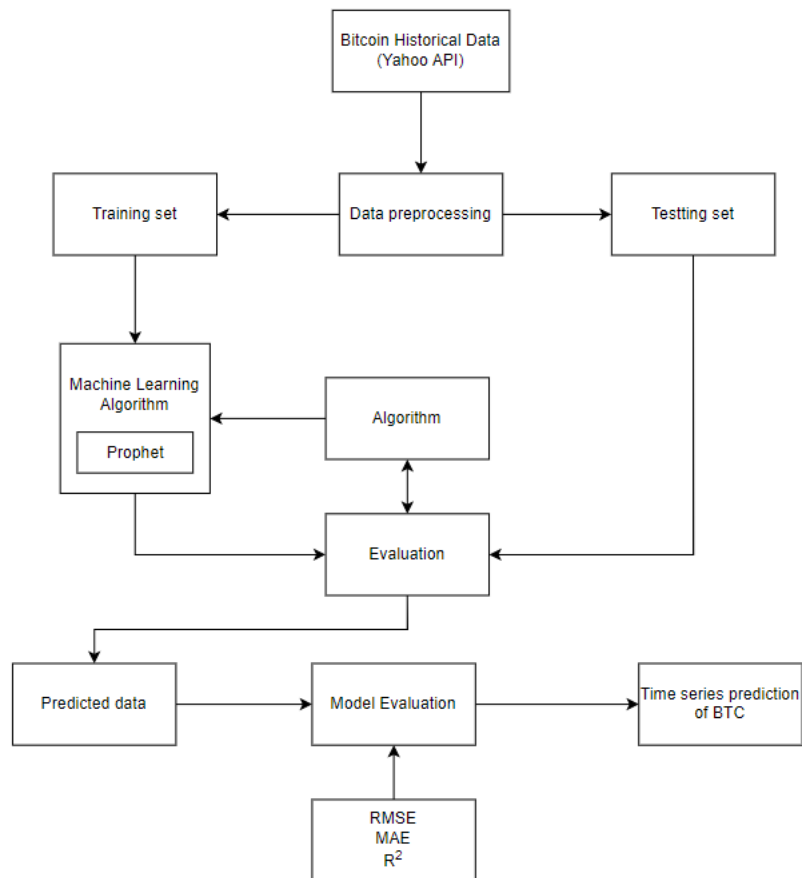
A minimum of 4GB RAM and 2GB memory is required to run prophet.

Operating System: Prophet is compatible with major operating systems such as Windows, macOS, and Linux.

Python: Prophet is implemented as a Python library and requires Python to be installed on the system. Prophet is compatible with both Python 2.x and 3.x versions.

Python Libraries: Prophet relies on several popular Python libraries, including NumPy, Pandas, and Matplotlib. Ensure that these libraries and their dependencies are installed.

4. FLOWCHART



5. RESULT

We were able to obtain time series forecasting model using prophet for a period of the next 30 days. The model evaluation was done based on metrics such as: MSE, RMSE, MAE, MAPE, MDAPE and SMAPE.

6. ADVANTAGES & DISADVANTAGES

Advantages

- Prophet is a relatively easy-to-use library, even for beginners.
- Prophet has been shown to be very accurate at forecasting time series data.
- Prophet can automatically detect and incorporate seasonality components, such as weekly or yearly seasonality, without requiring manual specification. This simplifies the modeling process and reduces the need for extensive feature engineering.
- Prophet has built-in capabilities to handle missing data and outliers, making it more robust to data imperfections compared to some other time series forecasting models. It can impute missing values and adjust for outlier impact during the modeling process.

Disadvantages

- While Prophet can capture some nonlinear patterns, it may struggle to capture more complex nonlinear relationships that may be present in Bitcoin price data. If the relationship between Bitcoin prices and other variables is highly nonlinear,
- Prophet requires historical data to train a model. If there is not enough historical data available, it may not be possible to make accurate predictions.
- Prophet's predictions can be sensitive to the choice of parameters. This means that it is important to tune the parameters carefully in order to get accurate predictions.
- Prophet can be computationally expensive to train and forecast. This means that it may not be feasible to use Prophet for forecasting large time series data sets.

7. APPLICATIONS

Apart from using prophet for cryptocurrency time series forecasting, it can also be used for:

1. Time series forecasting with Prophet can be used to predict financial variables such as stock prices, exchange rates, or commodity prices. It helps investors, traders, and financial institutions make informed decisions and develop trading strategies based on accurate forecasts.
2. Businesses can use Prophet for forecasting future demand for their products or services. This helps optimize inventory management, production planning, and

resource allocation. Retailers, e-commerce platforms, and supply chain companies often utilize time series forecasting to anticipate customer demand patterns.

3. Time series forecasting with Prophet is useful for predicting future sales volumes in retail, e-commerce, and other industries. It assists in demand planning, inventory management, and sales strategies.
4. Prophet can be applied to time series data related to weather variables, such as temperature, precipitation, or wind speed. Accurate weather forecasts are vital for various sectors, including agriculture, transportation, and renewable energy, enabling better decision-making and risk management.

8. CONCLUSION

In conclusion, time series forecasting using Prophet offers a powerful and intuitive approach to predict future values and patterns in time-dependent data. Prophet, addresses several challenges in time series analysis, such as handling seasonality, incorporating additional regressors, and providing interpretable results.

Its flexibility, ease of use, and ability to handle multiple seasonality patterns make it a popular choice for time series analysis and forecasting.

9. FUTURE SCOPE

The future scope of time series forecasting using Prophet is very promising. As Prophet continues to evolve, it is likely that we will see even more innovative and creative applications of this powerful tool. Here are some of the potential future applications of Prophet:

- Prophet is currently limited to forecasting time series that exhibit simple patterns. However, as Prophet continues to evolve, it is likely that we will see it being used to forecast more complex time series, such as those that exhibit non-linear trends or seasonality.
- It is likely that we will see Prophet being used to forecast multivariate time series.
- Time series forecasting using Prophet can benefit from advancements in big data technologies and real-time analytics. Integrating Prophet with scalable frameworks and platforms designed for processing and analyzing large-scale time series data can enhance its capabilities for handling massive datasets and enabling real-time forecasting applications.

10. BIBLIOGRAPHY

References of previous works or websites visited/books referred for analysis about the project, solution previous findings etc.

<https://github.com/facebook/prophet>

<https://github.com/ranaroussi/yfinance>

<https://www.mdpi.com/1911-8074/16/1/51>

<https://eudl.eu/pdf/10.4108/eai.7-7-2021.170286>

APPENDIX

Source Code:

https://colab.research.google.com/drive/1JW80nGmXSIhVmE0MQggoaavp_Oa1Tyap?usp=sharing

Github repo link:

https://github.com/deeppss/SmartInternz_Project/tree/main