JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Bitcoin price prediction Using Machine Learning

[*1]G.Preethi, [*2]L.Abdul Azeez, [*3]Dr.S.Jansi
[*1&*2]PG Scholars, [*3]Assistant Professor
[*1,*2&*3]Department of Computer Applications
[*1,*2&*3]Madanapalle Institute of Technology & Science, India

Abstract:

The purpose of this study is to see how well the direction of the Bitcoin price in USD can be anticipated. The Bitcoin Price Index is used to calculate the price. Through implementation, the task is accomplished with varied degrees of success. The Random Forest achieves the maximum accuracy in classification. Finally, the deep learning models are tested on a GPU and a CPU, with the GPU training time outperforming the CPU implementation by 67 percent.

Keywords: Deep Learning, Recurrent Neural Network, and Random Forest and Bitcoin.

Introduction:

Bitcoin is the most valuable crypto currency in the world, and it is traded on over 40 exchanges across the world that accept over 30 other currencies. According to https://www.blockchain.info/, it has a current market valuation of $9 billion USD and over 250,000 transactions every day. Due to its very short age and resulting volatility, Bitcoin as a currency presents a fresh potential for price prediction. It is also distinct from typical fiat currencies in that it is open; no complete data on cash transactions or money in circulation for fiat currencies available. The prediction of mature financial markets, such as the stock market, has received extensive attention.

Bitcoin is an interesting analogy because it is a time series prediction problem in a market that is still in its infancy. Traditional time series prediction approaches, such as Holt-Winters' exponential smoothing models, rely on linear assumptions and require data that can be divided into trend, seasonal, and noise components in order to be useful. This methodology is more suited to tasks including seasonal effects, such as sales forecasting. These strategies are not particularly effective for this work due to the absence of seasonality in the Bitcoin market and its extreme volatility. Given the task's complexity, deep learning appears to be an intriguing technological solution based on its performance in related areas.

The goal of this study is to investigate how accurately the Bitcoin price can be forecasted using machine learning and to compare parallelization approaches conducted on multi-core and GPU systems. This paper makes the following contributions: Only 7 (at the time of writing) of the approximately 653 papers published on Bitcoin are linked to machine learning for prediction. A Random Forest is also designed for performance comparison purposes in order to allow comparison to more traditional methodologies in financial forecasting.

1. Related works:

Bitcoin: A peer-to-peer electronic cash system: A peer-to-peer electronic cash system would allow internet payments to be transmitted directly from one party to another without going through a banking institution. Although digital signatures contribute to the solution, the main advantages are lost if a trusted third party is still necessary to prevent double-spending. We suggest a peer-to-peer network-based solution to the double-spending problem. The network hashes transactions into a continuing chain of hash-based proof-of-work, establishing a record that cannot be modified without redoing the proof-of-work. The longest chain serves as confirmation not only of the sequence of events witnessed, but also that it originated from the largest pool of CPU power. As long as nodes that aren't colluding to attack the network hold the bulk of CPU power, they'll build the longest chain and overtake attackers. The network itself demands little structure. Messages are broadcast using best efforts, and nodes can quit and rejoin the network at any time, accepting the longest proof-of-work chain as verification of what occurred while they were gone.

Virtual currency, tangible ` return: Portfolio diversification with bitcoins: Bitcoin is a well-known digital money. We examine a Bitcoin investment from the perspective of a U.S. investor with a diverse portfolio that includes both traditional assets (global stocks, bonds, and hard currencies) and alternative investments, using weekly data from 2010 to 2013. (commodities, hedge funds, real estate). Bitcoin investment has very distinguishing characteristics over the time period under examination, including exceptionally high average returns and volatility. It had a minimal correlation with other assets. Extensive testing confirms that Bitcoin investment provides substantial diversification benefits. We show that even a small percentage of Bitcoins in a well-diversified portfolio can significantly enhance the risk-return trade-off. However, results should be interpreted with caution because the data may reflect early-stage behaviour that does not last in the medium or long term.

Designing a neural network for forecasting financial and economic time series: Artificial neural networks are function approximators that are ubiquitous and very flexible. They were first employed in cognitive science and engineering. In recent years, the use of neural networks in finance for tasks such as pattern recognition, classification, and time series forecasting has grown substantially. However, due to the vast number of parameters that must be chosen in order to construct a neural network forecasting model, the design process still includes a significant amount of trial and error. The goal of this paper is to provide a practical first-step guidance to designing a neural network for forecasting economic time series data. An eight-step approach for designing a neural network forecasting model is described, together with a discussion of parameter tradeoffs, typical mistakes, and areas of dispute among practitioners.

Economic prediction using neural networks: The case of ibm daily stock returns: A report on the findings of an ongoing study that use neural-network modelling and learning approaches to search for and decode nonlinear regularities in asset price movements is presented. The author concentrates on the daily returns of IBM common stock. Dealing with the prominent aspects of economic data emphasises the relevance of statistical inference and necessitates adjustments to normal learning techniques that may be beneficial in other situations.

Holt-winters forecasting: some practical issues: Some practical issues in applying the method are explored, including seasonal indices normalisation, starting value selection, and smoothing parameter selection. There is a significant difference between an automatic and a non-automatic approach to forecasting, and specific recommendations for adopting Holt-Winters in both modes are provided. The method also addresses the question of what underlying model, if any, is assumed. Following that, several potential areas for future investigation are suggested.

Bitcoin academic paper database: Web of Science Core Collection provided the 887 documents used in the study. We created bibliometric maps based on text and bibliographic data using the VOSviewer software. Our research produces a knowledge area map that finds and evaluates the interconnections between authors and country distribution, the field's conceptual framework, and the structure and connections of the most cited publications and journals. To summarise our findings, we see a concentration of interest in certain keywords (Bitcoin, crypto currency, blockchain) and in some notable authors (with more than 100 citations per article). The research on Bitcoin as an economic concept accounts for only 33.5 percent of the total contributions in the field as a pure expression of digital economy.

2. Methodology:

Proposed system:

We propose this application as a valuable system because it assists in reducing the constraints gained by Random Forest. It can generate best outcomes for features with little overlap by offering support through forecasting analysis.

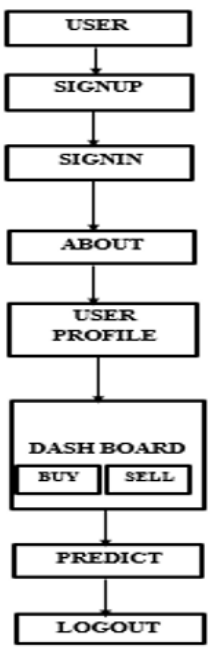


Figure 1: Block diagram of proposed method

3. Implementation:

The project has implemented by using below listed algorithm.

Random Forest:

A random forest is a machine learning technique for solving regression and classification problems. It makes use of ensemble learning, a technique that combines several classifiers to solve complex problems.

A random forest algorithm is made up of several decision trees. The random forest algorithm's 'forest' is trained via bagging or bootstrap aggregation. Bagging is a meta-algorithm ensemble that increases the accuracy of machine learning algorithms.

The outcome is determined by the (random forest) algorithm based on the predictions of the decision trees. It forecasts by averaging or averaging the output of various trees. Increasing the number of trees improves the outcome's precision.

The constraints of a decision tree algorithm are removed by using a random forest. It reduces dataset overfitting and boosts precision. It makes predictions without requiring extensive package parameters (like Scikit-learn).

• It is more accurate than the decision tree algorithm.

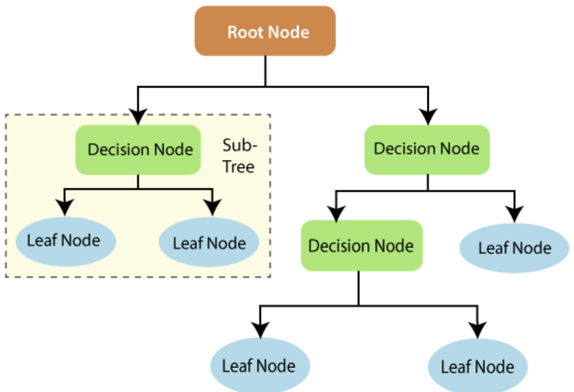
• It can generate an acceptable prediction without hyper-parameter tuning.

• It eliminates the problem of overfitting in decision trees.

A random forest algorithm is built from decision trees. A decision tree is a technique for decision assistance that has a tree-like structure. An overview of decision trees will assist us in comprehending how random forest algorithms operate.

A decision tree is made up of three different parts: decision nodes, leaf nodes, and a root node. A decision tree method separates a training dataset into branches, which then divide into further branches. This pattern is repeated until a leaf node is reached. The leaf node cannot be further isolated.

The properties represented by the nodes in the decision tree are used to forecast the outcome. The decision nodes connect the leaves. The diagram below depicts the three kinds of nodes in a decision tree.



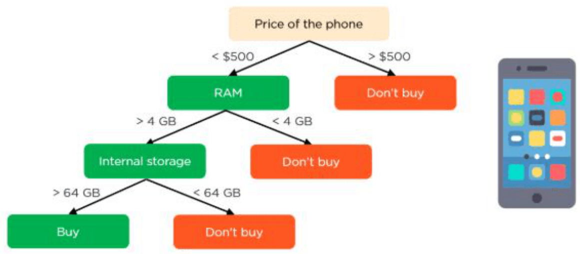
More information on how decision trees work can be found in information theory. Decision trees are constructed using entropy and information gain. An exploration of these key principles will help us better comprehend how decision trees are constructed.

Entropy is a measurement of uncertainty. Given a set of independent variables, information gain is a measure of how much uncertainty in the target variable is decreased.

The notion of information gain entails employing independent variables (features) to learn about a target variable (class). The information gain is estimated using the entropy of the target variable (Y) and the conditional entropy of Y (given X). The conditional entropy is removed from the Y entropy in this scenario.

In the training of decision trees, information gain is used. It contributes to the reduction of uncertainty in these trees. A substantial information gain indicates that a significant amount of uncertainty (information entropy) has been eliminated. Splitting branches, a key activity in the creation of decision trees, is influenced by entropy and information gain.

Let's look at a simple decision tree. Assume we want to forecast whether a customer would buy a mobile phone or not. His decision is based on the features of the phone. A decision tree diagram can be used to depict this study. The above-mentioned phone features are represented by the decision's root node and decision nodes. The final output, either buying or not buying, is represented by the leaf node. The pricing, internal storage, and Random Access Memory are the primary factors that influence the decision (RAM). The following is the decision tree.



Random forest decision trees

The fundamental difference between the decision tree method and the random forest algorithm is that the latter randomly establishes root nodes and separates nodes. To generate the required forecast, the random forest uses the bagging approach.

Bagging entails using multiple samples of data (training data) instead of simply one. A training dataset is made up of observations and attributes used to make predictions. Depending on the training data provided to the random forest method, the decision trees generate varied results. The best of these results will be picked as the final result.

Our original example can still be utilised to demonstrate the operation of random forests. The random forest will have several decision trees instead of a single decision tree. Let's pretend we just have four decision trees. In this example, the training data will be separated into four root nodes based on the phone's observations and features.

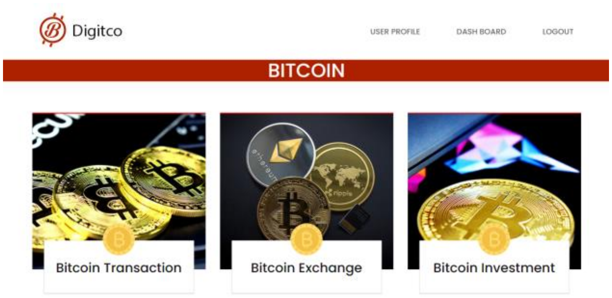
The root nodes may symbolize four characteristics that may influence the customer's decision (price, internal storage, camera, and RAM). The random forest will divide the nodes by randomly selecting features. The outcome of the four trees will be used to make the final prediction.

Most decision trees will choose the last option. If three trees indicate purchasing and one tree predicts not buying, then buying will be the final forecast. The customer is expected to purchase the phone in this situation.

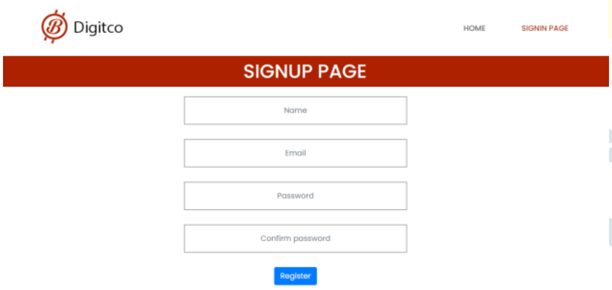
4. Results and Discussion:

The following images will visually depict the process of our project.

Home page: This is the home page of this project, it like a brief introduction of the project.

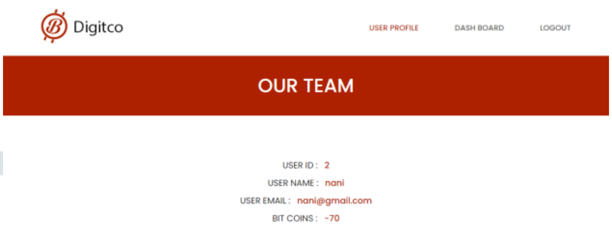


Sign up Page:

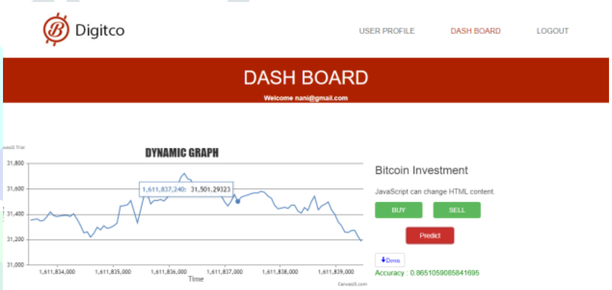


Sign in page:

Profile Page:



Dash Board:



5. Conclusion:

We have successfully developed a model to predict future outcomes for the digital currency Bitcoin in this application. Using Python programming, this is created in a user-friendly environment.

6. References:

1. Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," published in 2008.
2. M. Briere, K. Oosterlinck, and A. Szafarz, "Virtual money, tangible'return: Portfolio Diversification with Bitcoins," Tangible Return: Portfolio Diversification with Bitcoins (September 12, 2013), 2013.
3. I. Kaastra and M. Boyd, "Designing a neural network for forecasting financial and economic time series," Neurocomputing 10, no. 3, pp. 215–236, 1996.
4. H. White, "Economic prediction using neural networks: The case of IBM daily stock returns," IEEE International Conference on Neural Networks, 1988, pp. 451–458.
5. C. Chatfield and M. Yar, "Holt-winters forecasting: some practical concerns," The Statistician, vol. 129, no. 4, 1988, pp. 129–140.
6. suitpossum blog, B. Scott, "Bitcoin academic article database," 2016.

7. "Machine-learning classification algorithms for the analysis and prediction of high-frequency stock direction," M. D. Rechenthin, 2014.
8. D. Shah and K. Zhang, "Bayesian regression and bitcoin," in Communication, Control, and Computing (Allerton), IEEE, 2014, pp. 409–414.
9. G. H. Chen, S. Nikolov, and D. Shah, "A latent source model for nonparametric time series classification," Advances in Neural Information Processing Systems, vol. 8, no. 3, 2013, pp. 1088–1096..
10. I. Georgoula, D. Pournarakis, C. Bilanakos, D. N. Sotiropoulos, and G. M. Giaglis, "Detecting bitcoin price drivers using time series and sentiment analysis," 2015, available at SSRN 2607167.
11. M. Matta, I. Lunesu, and M. Marchesi, "Bitcoin spread prediction utilising social and web search media," DeCAT 2015 Proceedings.
12. "The impact of web search media on bitcoin trading volumes as a predictor."
13. B. Gu, P. Konana, A. Liu, B. Rajagopalan, and J. Ghosh, "Identifying information in stock message boards and its implications for stock market efficiency," in 2006 Workshop on Information Systems and Economics, Los Angeles, CA.
14. A. Greaves and B. Au, "Using the Bitcoin Transaction Graph to Forecast Bitcoin Price," 2015.
15. I. Madan, S. Saluja, and A. Zhao, “Automated bitcoin trading via machine learning algorithms,” 2015.