Can a hybrid of ANN-GARCH model provide a significant improvement in predicting the price volatility of the Ethereum?

MSc Research Project
FinTech

Sagar Anil Tade
Student ID: X18109641

School of Computing
National College of Ireland

Supervisor: Noel Cosgrave

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Sagar Anil Tade
Student ID:	X18109641
Programme:	FinTech
Year:	2019
Module:	MSc Research Project
Supervisor:	Noel Cosgrave
Submission Due Date:	12/08/2019
Project Title:	Can a hybrid of ANN-GARCH model provide a significant improvement in predicting the price volatility of the Ethereum?
Word Count:	4192
Page Count:	14

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	12th August 2019

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Can a hybrid of ANN-GARCH model provide a significant improvement in predicting the price volatility of the Ethereum?

Sagar Anil Tade
X18109641

Abstract

Ethereum the second most popular cryptocurrency next to Bitcoin has a market share of US $18 billion. Due to its huge market shares, Ethereum has gained attention from the investor community. With its unique ability to write smart contracts for blockchain technologies, it is very popular among the application developers community. Even after having a significantly large fan base, there is very little academic research being done on this cryptocurrency. This research paper aim towards finding out if the hybrid model consisting of ANN-GARCH can provide significant improvements in predicting the price volatility of such popular and volatile cryptocurrency. This research intends to contribute to a better understanding of the ability of the GJR-GARCH model to predict volatility with a rolling window approach. These results when fed to the LSTM model for training and closing the gap between the actual and forecasted volatility values of the GARCH model yield interesting results. Root Mean Squared Error (RMSE) of the GJR-GARCH model is 0.6986, RMSE of LSTM Model is 0.6071. Whereas, the RMSE for the hybrid model is 0.5894. This denotes that the hybrid model is good at reducing the error thus predicting more accurately than the standalone models. The question which remains after this research is if it worth having a hybrid model to increase the performance by 2% or wait for some more time for Ethereum to generate sufficient data points to train heavy models likes of LSTM to yield better results.

Contents

1 Introduction

Cryptocurrencies have started gaining huge popularity after the 2008 crisis. With more than US $180 billion being invested in the cryptocurrency markets it is becoming one of the major asset class. Increase in popularity has attracted every individual's attention towards the cryptocurrency market.[1] Stories of people becoming millionaires overnight are hyping up the interest of the investors and common people. The very high volatile and unregulated market of cryptocurrencies has enabled people in gaining huge profits in recent years. On the other hand, billions of dollars were lost due to the same characteristics causing huge risks to the investors and fund managers creating chaos. Thus various academicians and professionals are trying to solve the problem of predicting the volatility of such cryptocurrencies. Majority of the study is done across one cryptocurrency which is Bitcoin. Origin of Bitcoin was aimed towards overcoming the shortfalls and opaque systems of the traditional banking institutions. The underlying technology of Bitcoin which was termed as "Blockchain" had trust, privacy, transparency and integrity into it by design with the use of advanced maths and technology. Thus the underlying technology of Bitcoin was a great inspiration for other cryptocurrencies and Ethereum in particular.

Ethereum which is the second popular and second most traded cryptocurrency has a capitalization of US $18 billion. With similar underlying technology of blockchain, Ethereum is also used for developing applications based on blockchain technologies. Ethers can be traded for the computation power required to solve a complex math problem for verifying the result. Thus making it unique in itself. One can create its own blockchain application based on Ethereum blockchain thus allowing it to execute applications and transacting the payments. Keeping this unique feature of Ethereum in mind, it was necessary to have research with regards to the prediction of the volatility of this cryptocurrency. Hybridized model of Artificial Neural Network (ANN) and Generalized Autoregressive Conditional Heteroskedasticity (GARCH) was thought of to be best suited for this research. The ability of GARCH models to estimate the volatility of an asset is the central idea of using it. Whereas, ANN is used to detect and minimize the nonlinear effects of volatility. (Kristjanpoller and Minutolo; 2018)

This led to the proposal of the research question of: Can a hybrid of ANN-GARCH model provide a significant improvement in predicting the price volatility of the Ethereum?

To answer this question, the following milestones were set as objectives.

1. To statistically validate if the Ethereum time series is competent for the use of a proposed hybrid model.

2. To implement a rolling window approach in predicting the volatility by using GARCH and ANN.

3. To compare the hybrid results with the traditional systems and highlight its significance.

[1] CoinMarketCap: `https://coinmarketcap.com/`

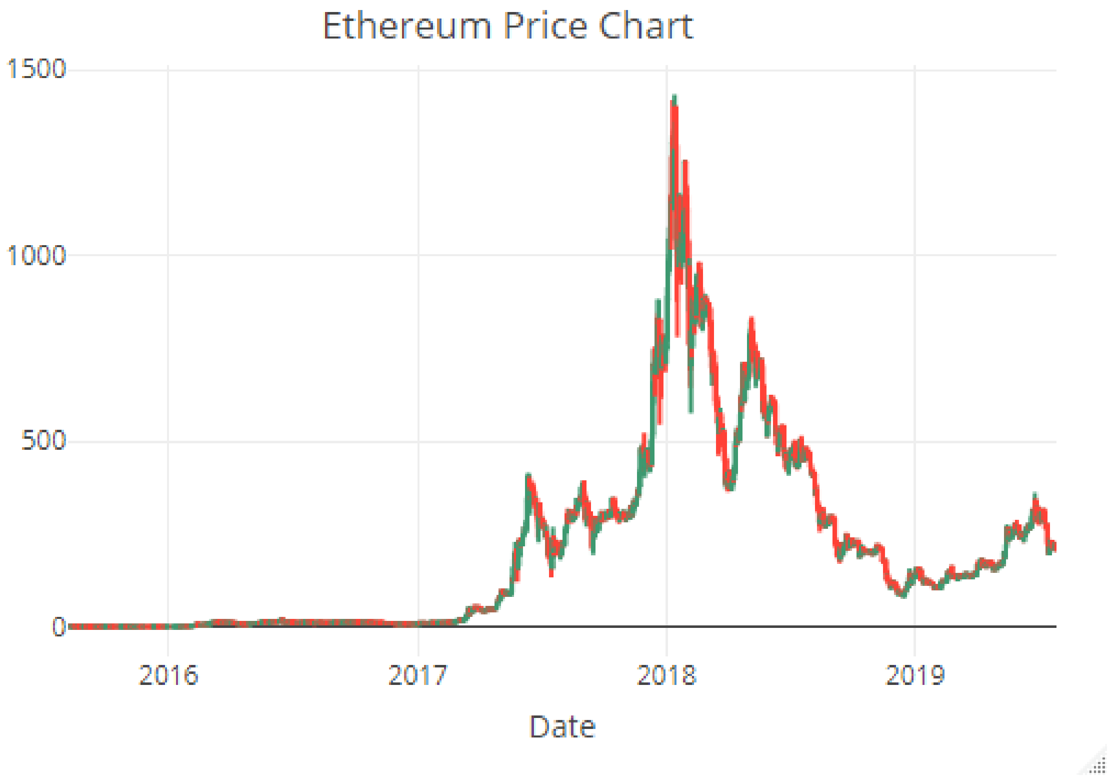


Figure 1: Ethereum Price Chart Candlestick

This research is done only on the basis of historical time series data without any consideration of external factors or news or change in governing policies by any influential countries. Any of the above-mentioned factors may imply in a sudden change of prices and volatilities which may be the limitation of this research Along with this there are only 1451 data points available for this research which turn out to be insufficient for a heavy model like LSTM to train with. It is assumed that the current state of cryptocurrencies continue in the future as they are now. Moving forward with these assumptions, the next section of the report will include a brief review of the literature in the area of cryptocurrencies, volatility predictions using hybrid models and state of the art in the Ethereum price prediction. The proposed methodology to achieve the answers to the questions listed will be discussed after the review based on the literature. Next sections of this research paper will focus in detail on the implementation part following which will be a detailed discussion of the results obtained and how are they interpreted. Thus concluding this research and highlighting the possible future work which can add more value to the current research.

2 Related Work

The cryptocurrency market is very loosely regulated. It has been in existence a little over a decade. Hence there is very little literature surrounding it. Most of the academic literature which is available these days is surrounding the popular cryptocurrency Bitcoin (Böhme et al.; 2015). The research conducted around this space speaks about how Bitcoin can be used for conducting illegal activities. Another research conducted by the researchers from China talks about the concerns of the capital movement towards cryptocurrencies from financial markets in China. (Ju et al.; 2016)

On the other hand, academicians talk about how investing in Bitcoin or other cryptocurrencies can help to make an investment portfolio stronger and how it has created new opportunities for investment (Briere et al.; 2015). But the major limitation of their research is that it was conducted when the prices were skyrocketing thus failing to detect and understand the negative shocks. To predict returns of Bitcoin by including it in the portfolio of fiat currencies, one research uses the GARCH(1,1) and GARCH(1,2) (Bhattacharjee; 2016). Though their results were interestingly stating the fact that Bitcoin forecasted better returns than the fiat currencies which they were compared across.

The uniqueness of Ethereum as a cryptocurrency is discussed in the research paper (Wang et al.; 2019). They highlight that the use of Ethereum for smart contracts has a sizable amount of funds and a huge number of accounts for trading it in the market. They also state the fact that being a cryptocurrency, Ethereum has very similar characteristics as that of Bitcoin.

To support the above research, the white paper on Ethereum (Buterin et al.; 2014) describes how it is an updated version of other cryptocurrencies with unique features such as creating financial contracts. Being an open-ended protocol, allowing people to develop applications and transacting the cryptocurrency for executing it is also possible. Taking all these factors into account, prices of Ethereum are not only based on economic factors alone thus making it hard to predict.

Since GARCH Models and its variations are good at predicting such kind of volatile assets, they were chosen for this research. One of the research papers has compared twelve GARCH models across four cryptocurrencies. They discovered that GJR-GARCH(1,1) and IGARCH(1,1) were the best models for the majority of the cases. GARCH(1,1) was best fit for Ripple cryptocurrency. Best fitting models were discovered based on the AIC, BIC and Hannan Quinn Criterion (Chu et al.; 2017). Lower the value of the criterion, better is the GARCH fitting likelihood. Surprising enough this research excluded the Ethereum from its cryptocurrencies invoking more curiosity for the research. This research helps in understanding that even after being under the umbrella, every cryptocurrency behaves differently.

The second part of the research is to implement Artificial Neural Network (ANN) to reduce the error component from the obtained GARCH volatility. Similar kind of research is being around for more than a decade. First spotting of use of ANN for predicting the stock prices as in the late '90s. It was carried out on the stock price of IBM. The results obtained then were not that impressive, but it provided the required motivation for tuning the model to train and achieve better results (White; 1988). Later research showed that a combination of GARCH and Neural Network was way more capable of capturing the volatility accurately rather than GARCH alone. Thus motivating the research thereafter (Kryzanowski et al.; 1993).

Research done on the Istanbul Stock Exchange in the year 2009 was one of the greatest motivations for carrying out this research. Data of the ISE 100 Index for this research was considered from October 1987 to February 2008. This data was nonlinear in nature and also has asymmetric clustering of volatility. These characteristics are very much similar to that of Ethereum. After using a plethora of models, the researchers concluded that the combination of APGARCH model with ANN turned out to be best (Bildirici and Ersin; 2009).

The recent studies of 2018 proposed the use of Long Short Term Memory which is a type of Recurrent Neural Network for predicting the volatility of Bitcoin. The research used a similar kind of framework that is being proposed in this paper. Researchers used the AR(2) model in association with LSTM. This research is very vital in terms to achieve our goals and objectives for the research (Wu et al.; 2018).

So it can be observed that as the timeline shifts to more recent studies, it is found that the researchers are very keen and interested in opting for a hybrid model as of traditional models. As a result, the hybrid model tends to provide better results in terms of accuracy for predictions. Especially the ANN and its variation LSTM is being used to reduce the gap between the forecasted and actual values of Bitcoin and other highly volatile markets.

3 Methodology

This research follows the CRoss Industry Standard Process for Data Mining which is known as CRISP-DM. It is an industry standard which helps to follow a particular order to achieve the project outcome in data analytics field systematically without missing or skipping any steps (Wirth and Hipp; 2000).

3.1 Business Understanding:

As discussed in the above sections, Ethereum is a unique cryptocurrency in itself. Predicting its prices and volatility is a major challenge in the current time. As there is significantly little work done in predicting its prices it shall be very useful for the investors. Academicians could also know the effects of using a hybrid model for predictions of such highly volatile asset class and know if it is efficient as compared to the traditional data modeling techniques.

3.2 Data Understanding:

Time series dataset of Ethereum prices is used. This data is scraped from the CoinMarketCap [2]. Data was available from 07th August 2015. For this particular research last date considered was 27th July 2019. It consists of 1451 observations and 5 attributes namely Date, Open, High, Low, Close. Frequency of this data is daily. Prices mentioned are across US Dollars. 2 Below is the decomposition summary image of the closing prices of Ethereum which is used for this research. It consists of observed, trend, seasonality and randomness in the time series. It suggests the time series had a upward trend till early 2018, thereafter its a declining trend. The randomness during the same time period is very high in this time series indicating sudden changes in he volatility. It also suggests that the time series does not follow any uniform trend and seasonality with it.

3.3 Data Preparation:

It is a vital step in this research. As the data obtained is scraped from online sourcesec:Introduction, it is raw data. So the next step is to convert this data into a time-series format for further use. Closing prices of this time series were then selected. A new column is then added to this time series. This new column has the calculated log values of the closing prices. Log values are fed to the models for training the model. Next column which is added to

[2] CoinMarketCap: `https://coinmarketcap.com/`

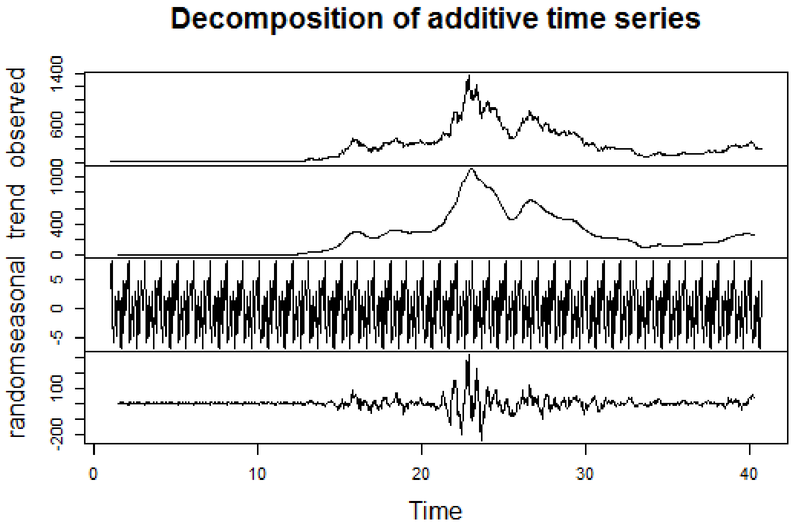


Figure 2: Ethereum Price Chart Candlestick

this time series has a difference between current and previous log price. This step was essential for deriving the standard deviation of the corresponding price. As the sliding window approach is adopted for this research, it is necessary to calculate the standard deviation of these lagged log values. A sliding window of 30 observations is decided for this research. Hence the next column which is being added to this time series dataset is of the sliding window standard deviations of the previous 30 observation. The final step in the data preparation is to calculate the volatility of the closing prices. Yang-Zhang estimator for calculating volatility is used as it is independent of the jumps and the drifts in the time series and is also has the assumption that the pricing is continuous which is true in case of Ethereum. Its maximum efficiency is 14 times better than that of close to close estimator (Yang and Zhang; 2000).

3.4 Data Modeling:

Before proceeding towards the modeling few tests are needed to be done on the volatility derived. These tests will give us the clear idea if the data obtained can be used for the statical modeling. In order to run the GARCH Model, following tests were essential.

The first test which is carried out on the Volatility is the Augmented Dickey Fuller (ADF) test. It is used for checking if the series is stationary or not. The null hypothesis for this test is that the series is not stationary and therefore consists of a presence of unit root. Whereas, the alternative hypothesis is that the series exhibits stationarity. The more the negative value higher is the chance of missing unit root. We can only proceed with this time series if is stationary. On applying this test it was found that the value is -4.2248 and the p-value is 0.01 thus rejecting the null hypothesis and stating that the time series may be stationary (Fuller; 2009).

To confirm the trend stationarity exists and support the ADF, Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test is also used. As opposite to the ADF test, null hypothesis of KPSS states that the series is stationary with deterministic mean that is the series is trend stationary against the null hypothesis stating that the different type of stationarity exists in the time series. As the LM statistic is less than the critical value of 0.05 confidence of 0.463 and the p-value is less than 0.01 states that the null hypothesis is accepted and the series is stationary (Shin and Schmidt; 1992).

Engle proposed Lagrange Multiplier test which is commonly termed as LM ARCH Test. This test involves fitting of a linear regression model for the squared residuals and then determining whether or not the fitted model is significant (Engle; 1982). When tested for ARCH presence in Ethereum time series, the null hypothesis stating the absence of ARCH process is rejected. Thus accepting the alternative hypothesis of q degrees of freedom greater than the squared value of X from the distribution and stating the presence of ARCH effect in the time series. The ARCH model is suitable when the time series with error variance follows the autoregressive (AR) model. Similarly, when autoregressive moving average (ARMA) is considered for the error variance, it is then termed as generalized autoregressive conditional heteroskedasticity (GARCH) model.

Figure below 3 shows the architectural design of the proposed research. Two main algorithms used here are GJR-GARCH(1,1) and LSTM.

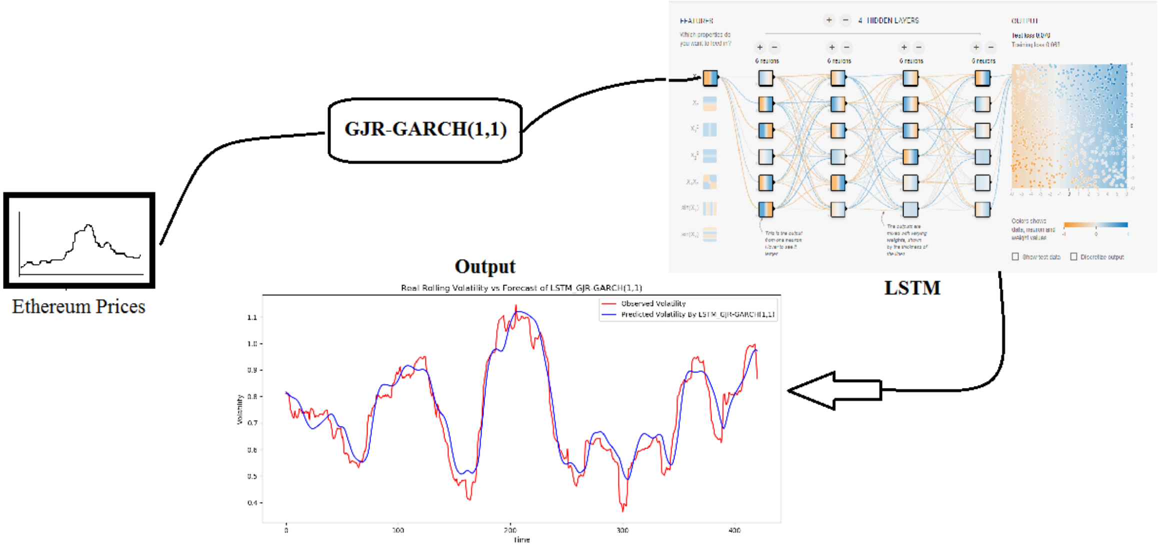


Figure 3: Architecture Diagram

After testing for ARCH presence, moving forward is finding the best GARCH model for this time series. There are total of 11 generalized autoregressive conditional heteroskedasticity (GARCH) models. After, thorough analysis of these models, 3 models are shortlisted for this research. They are EGARCH, QGARCH and GJR-GARCH. Starting with EGARCH model, as GARCH models ignores the sign of the variance thus modeling only magnitude. EGARCH over comes this issue considering negative signs from standard deviation. EGARCH captures the negative shock from the previous time shock t-1 and have a stronger impact for current time t rather than positive shocks (Nelson and Cao; 1992). But as the actual time series consists both positive and negative shocks, hence this model was discarded. Second model chosen was QGARCH. Like EGARCH, Quadratic GARCH model can detect the asymmetric positive as well as negative shocks. Addition of the asymmetric term to the vanilla GARCH improved the model significantly. It uses time-varying conditional variances for doing it (Sentana; 1995). According to the paper (Glosten et al.; 1993), GJR-GARCH is the modification of the GARCH-M model.

Three main considerations of GJR-GARCH are the seasonal volatility pattern, different impact of positive and negative impact on conditional variance and better prediction of conditional variance. Thus this model is chose for this research as it covers variety of the conditions which a highly volatile time series such as Ethereum can posses. Detail and configuration of this model is discussed in the following section of design specification.

Second tough choice is to select the appropriate artificial neural network model to reduce the error and train the model. As neural networks does not make any assumptions about the mapping function and promptly learn the linear and nonlinear relations between the data points. Neural networks are also very robust to the error from the input data regardless of the missing values of outliers or sudden changes (Tealab et al.; 2017). By keeping these factors in mind, Multilayer perceptron which is a feed forward type of neural network was sought after. Such kind of neural network are capable in solving most of the classification and non linear types of problems. But, the problems regarding time dependence is its limitation. To overcome this problem, recurrent neural networks (RNN) is a better solution. Among all the RNN's Long Short Term Memory (LSTM) takes into consideration the previous lagged training data and trains the model accordingly. When learning a mapping function, LSTM handles the order from inputs to outputs. Mapping function of LSTM learns to map inputs to outputs over a time period. This capability of LSTM is very useful for this research as volatility of Ethereum is varying widely across the time period t. LSTM thus helps this research by minimizing the error between the predicted volatilities and actual volatilities by using fixed size time windows (Gers et al.; 2002).

4 Implementation

GJR-GARCH(p,q) model is built where value of p=1 and q=1. Scaled volatility values are then passed through this model for fitting. To generate the rolling forecast volatility, alpha, beta, gamma and omega values obtained from the GJR-GARCH model fitting are required. These values are fetched from the summary of the fitting. Next step in implementation is to solve the GJR-GARCH for each available record and obtain its forecasted volatility based on its t-1 lag value. Surprisingly when the scaling of the volatility was changed to 0.3 instead of 0.1, the modle fit was excellent. The RMSE at that scaling was 0.49. But as the reason for this phenomenon was not known, values were scaled back to the original volatility. Thus the GJR-GARCH model is denoted by the formula,

$$\sigma_t^2 = \omega + \ \alpha \ \epsilon_{t-1}^2 + \ \beta \ \sigma_{t-1}^2 + \ \gamma \ \epsilon_{t-1}^2 I_{t-1}$$

Where,

$$\epsilon_t = \ \sigma_t z_t \text{ and } I_{t-1} = 0$$

$$\text{if, } \ \epsilon_{t-1} \geq 0$$

$$\text{and } I_{t-1} = 1 \text{ if } \ \epsilon_{t-1} < 0$$

Thus forecasted GJR-GARCH values were plotted against the actual values for visual comparison. Below is the figure 4 showing the forecasted volatility across the actual volatility of Ethereum.

Next step involved in implementation phase is to create a training data set for LSTM model with 1121 observations, 1 input layer with 1 node and 1 output for it. LSTM is a type of Recurrent Neural Network. So it can send feedback connection to the existing

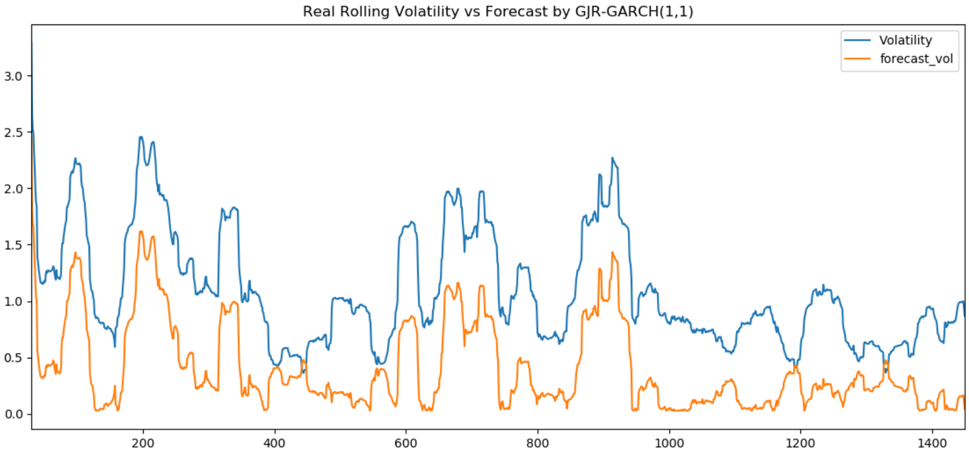


Figure 4: GJR-GARCH Forecast

node. Thus making it more complex. The architecture used for this research is pretty straight forward. it has an forget, output gate and input gate. Model consisting of the volatility prices is then fed to this feedback network. Features used for this particular research are, LSTM with 4 layers, 10 nodes per layer and 0.1 dropout for each layer to regularize. 5th layer was the output layer with 1 node. RNN regressor of tensorflow is used for implementation with 100 epochs and 32 batch size. For dataset of 1421 records, 100 epochs seemed to be optimum based on trial and error. Also the neurons are equally weighted for this research. Figure 5Figure below shows the visual representation of the actual vs predicted rolling LSTM volatility. This research is using ”Adam” optimizer for stochastic gradient descent thus optimizing ”mean squared error” as a loss function. Adam optimizer will change the weights of the nodes every iteration for the training dataset, making it much robust. It is considered better than the stochastic Optimization (Kingma and Ba; 2014). Its features such as efficient use of computing power, handles non stationary objects and minimum memory requirements makes it very likely to be used for this research. It is the most common and very effective optimizer. Also, as the Ethereum is volatile, total number of iterations that is epochs were fixed to 100. It takes around 37-40 sec for the dataset to traverse back and forth and complete one iteration that is epoch. To do this efficiently, dataset was subdivided into batches of 32 data points each. Thus taking around 1 hour for model fitting.

To reduce the error in the GJR-GARCH model forecasting, its forecasted volatility is fed to the LSTM model along with the normal volatility as a input. Hence the model have 2 input layers and 1 output layer while keeping rest of the parameters same.Addition on one more input node to the LSTM makes a model complexity higher. Thus this model takes around 40 sec to execute each epoch. is Moving forward to the next section which contains information about the evaluation of the results obtained.

5 Evaluation

First evaluation of this research will be the GJR-GARCH(1,1) Model. Below table 6 shows the summary of the model. It has all the details about the model which ran on

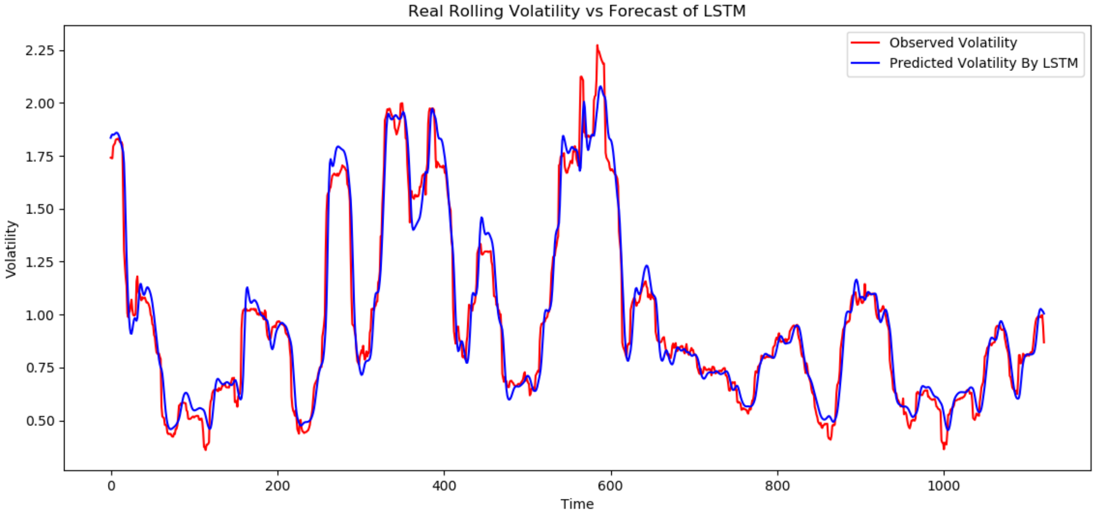


Figure 5: LSTM Forecast

the volatility estimator.

The mean model used is Constant Mean. The R-Squared value and the Adjusted R-Squared value is -0.252. The AIC Criterion is 101.657 where as the BIC Criterion is 127.953. These values denote the penalty for a model being complex to fit to the model. Thus stating its likelihood. BIC penalizes the model more rigorously with logarithmic scaling. We can understand that the Residuals are 1416 which are 5 less than the number of observations 1421. It also shows the log likelihood of the model which is -45.8285.Less the log-likelihood, better the model. As it is a negative number here, GJR-GARCH(1,1) is more likely a better fit for this dataset. The most important values from that summary for us are the coefficients omega, aplha, beta and gamma. Also the the number of residuals as each residual value will be fitted into the equation mentioned above of GJR-GARCH for volatility prediction. By using that we are calculating the rolling window forecasted frequency of GJR-GARCH(1,1) for each data point.

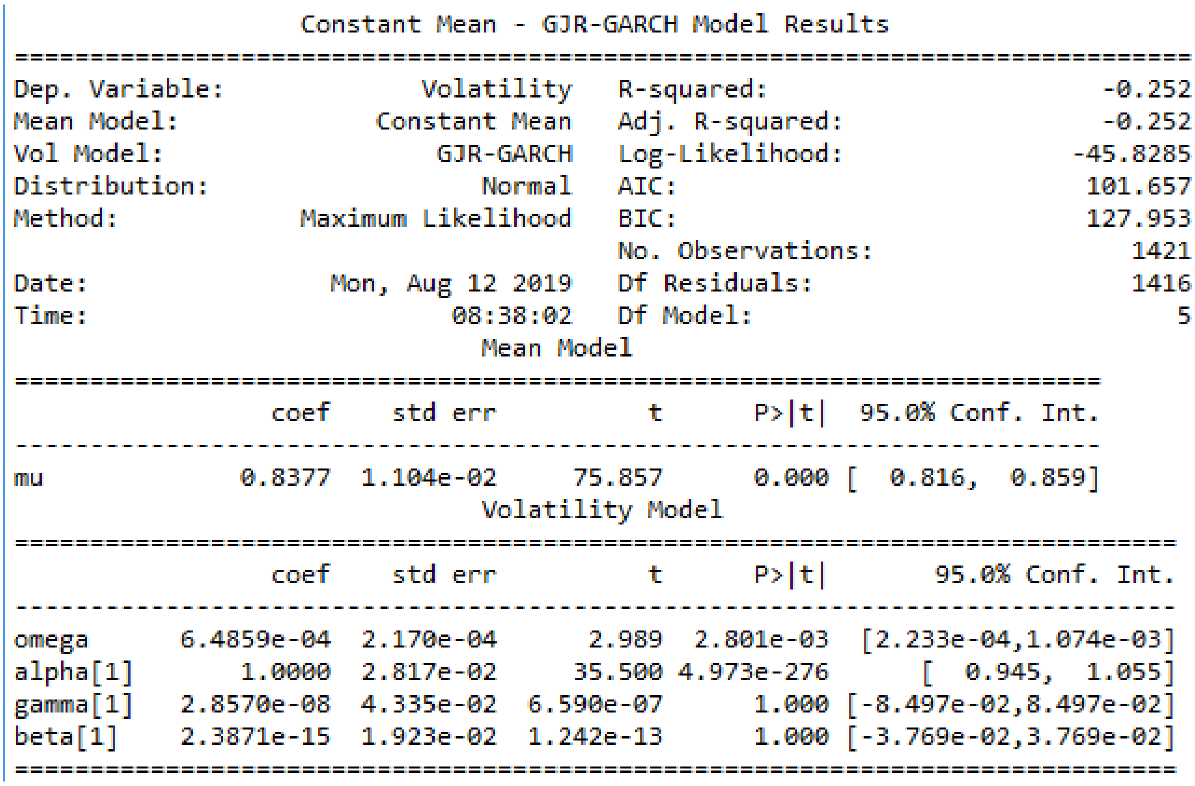
Next evaluation consists of finding the Root mean squared error between the actual and the predicted volatilities by the GJR-GARCH(1,1) model. The formula for which is

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{T}(\hat{y}_t - y_t)^2}{T}}.$$

Where y_t is the actual values and $\hat{y}_t$ are the predicted ones. This will be the measure for calculating the effectiveness of the model. In this case RMSE obtained was 0.6986

Next step involved using LSTM on the volatility to check what is its RMSE to check the effectiveness of LSTM on the forecasted volatility values. It was observed that the LSTM has mimicked the real data with lag. But even with this the RMSE obtained was 0.6071. RMSE value obtained from the LSTM is 13.09% better than that of the GJR-GARCH(1,1) Model.

For its final step, that is to verify if the hybrid model of ANN-GARCH provides a significant improvement in the volatility prediction, volatility values obtained from rolling window approach for GJR-GARCH(1,1) were passed along with the actual Volatility to train the model. Following results were obtained 7.



```
                   Constant Mean - GJR-GARCH Model Results                    
==============================================================================
Dep. Variable:             Volatility   R-squared:                      -0.252
Mean Model:             Constant Mean   Adj. R-squared:                 -0.252
Vol Model:                  GJR-GARCH   Log-Likelihood:               -45.8285
Distribution:                  Normal   AIC:                           101.657
Method:            Maximum Likelihood   BIC:                           127.953
                                        No. Observations:                 1421
Date:                Mon, Aug 12 2019   Df Residuals:                     1416
Time:                        08:38:02   Df Model:                            5
                               Mean Model                              
=======================================================================
                 coef    std err          t      P>|t|  95.0% Conf. Int.
-----------------------------------------------------------------------
mu             0.8377  1.104e-02     75.857      0.000 [  0.816,  0.859]
                               Volatility Model                              
=============================================================================
                 coef    std err          t      P>|t|       95.0% Conf. Int.
-----------------------------------------------------------------------------
omega      6.4859e-04  2.170e-04      2.989  2.801e-03  [2.233e-04,1.074e-03]
alpha[1]       1.0000  2.817e-02     35.500 4.973e-276      [  0.945,  1.055]
gamma[1]   2.8570e-08  4.335e-02  6.590e-07      1.000 [-8.497e-02,8.497e-02]
beta[1]    2.3871e-15  1.923e-02  1.242e-13      1.000 [-3.769e-02,3.769e-02]
=============================================================================
```

Figure 6: LSTM Forecast

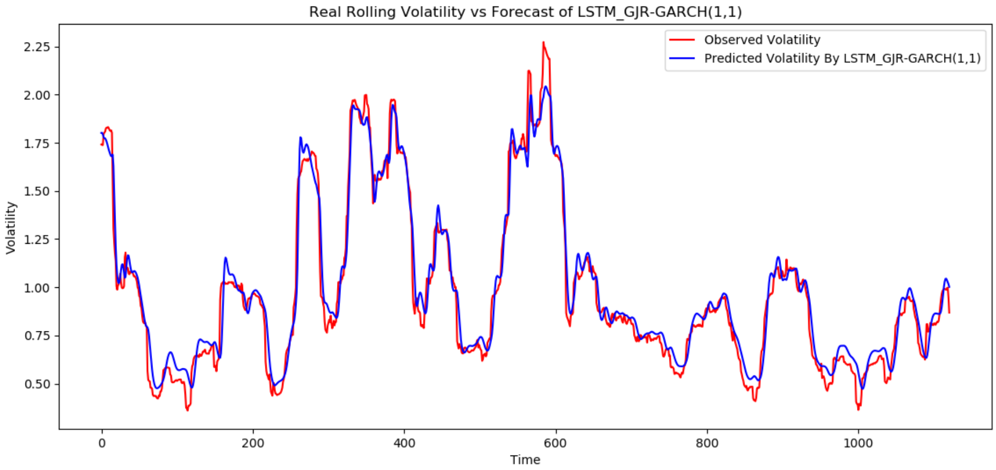


Figure 7: GARCH-LSTM

From the graphical representation of Figure 7, it seems like the model has generated the better results than standalone LSTM and GJR-GARCH(1,1). RMSE obtained for the same is 0.5894, which is a better indication. But the RMSE is improved only by 2.03

Table 1 below denotes RMSE values obtained by all the models.

Table

Table 1: Root Mean Squared Error.

Model	**RMSE**	%Change
GJR-GARCH(1,1)	0.6986146423	NA
LSTM	0.6070618624	13.09%
GARCH+LSTM	0.5894160298	2.03%

6 Conclusion and Future Work

Its safe to say that hybridization is trending now a days in data science. With the rise of high volatile asset classes such as cryptocurrency and use of cryptocurrencies not only for transacting money but also for building applications and running them, it is becoming a huge challenge for predicting the volatility. 1st objective set for this research to check if the Ethereum is competent for the use of proposed hybrid model. The answer is yes. It is proved by using ADF, KPSS and ARCH test. Which all approved the Ethereum to be eligible for the applying a hybrid model. Second objective to implement a rolling window approach in predicting the volatility was achieved by training the model on first 30 rows and then finding the volatility by Yang Zang Estimator which covers drifts as well as jumps. Coefficients of alpha, beta and gamma are separated to find the rolling volatility by using the estimation formula. Thus applying rolling volatility forecast with window size 30. The third and the final objective of comparing the results with traditional standalone model is also achieved. To compare the results Root Mean Squared Error measure of accuracy is used. The results obtained suggested that the hybrid model has the least error thus implying it is better suited hence answering the research question if a hybrid model of ANN-GARCH can provide significant improvement in predicting the price volatility with a answer as yes.

But we cannot ignore the fact that, the hybrid model is just over 2% better than that of the traditional LSTM model. Which rises the question, if it is worth it. This might have caused due to the less amount of data for a heavy model like LSTM to be trained. Also the limitation of this research is, it only consists of the historical data, but totally ignores external factors affecting the pricing and volatility such as news outbreak, change in governance, imposition of new rules, etc. It may really be interesting to see the results of sentiment analysis on such new and volatile asset class. This research can be revisited once there is adequate amount of data available for the training of heavy models like LSTM. Until then, this research can be used as a base for developing new hybrid models and testing their volatility.

References

Bhattacharjee, S. (2016). A statistical analysis of bitcoin transactions during 2012 to 2013 in terms of premier currencies: Dollar, euro and rubles, *Vidwat* **9**(1): 8.

Bildirici, M. and Ersin, Ö. Ö. (2009). Improving forecasts of garch family models with the artificial neural networks: An application to the daily returns in istanbul stock exchange, *Expert Systems with Applications* **36**(4): 7355–7362.

Böhme, R., Christin, N., Edelman, B. and Moore, T. (2015). Bitcoin: Economics, technology, and governance, *Journal of Economic Perspectives* **29**(2): 213–38.

Briere, M., Oosterlinck, K. and Szafarz, A. (2015). Virtual currency, tangible return: Portfolio diversification with bitcoin, *Journal of Asset Management* **16**(6): 365–373.

Buterin, V. et al. (2014). A next-generation smart contract and decentralized application platform, *white paper* **3**: 37.

Chu, J., Chan, S., Nadarajah, S. and Osterrieder, J. (2017). Garch modelling of cryptocurrencies, *Journal of Risk and Financial Management* **10**(4): 17.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation, *Econometrica: Journal of the Econometric Society* pp. 987–1007.

Fuller, W. A. (2009). *Introduction to statistical time series*, Vol. 428, John Wiley & Sons.

Gers, F. A., Eck, D. and Schmidhuber, J. (2002). Applying lstm to time series predictable through time-window approaches, *Neural Nets WIRN Vietri-01*, Springer, pp. 193–200.

Glosten, L. R., Jagannathan, R. and Runkle, D. E. (1993). On the relation between the expected value and the volatility of the nominal excess return on stocks, *The journal of finance* **48**(5): 1779–1801.

Ju, L., Lu, T. and Tu, Z. (2016). Capital flight and bitcoin regulation, *International Review of Finance* **16**(3): 445–455.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* .

Kristjanpoller, W. and Minutolo, M. C. (2018). A hybrid volatility forecasting framework integrating garch, artificial neural network, technical analysis and principal components analysis, *Expert Systems with Applications* **109**: 1–11.

Kryzanowski, L., Galler, M. and Wright, D. W. (1993). Using artificial neural networks to pick stocks, *Financial Analysts Journal* **49**(4): 21–27.

Nelson, D. B. and Cao, C. Q. (1992). Inequality constraints in the univariate garch model, *Journal of Business & Economic Statistics* **10**(2): 229–235.

Sentana, E. (1995). Quadratic arch models, *The Review of Economic Studies* **62**(4): 639–661.

Shin, Y. and Schmidt, P. (1992). The kpss stationarity test as a unit root test, *Economics Letters* **38**(4): 387–392.

Tealab, A., Hefny, H. and Badr, A. (2017). Forecasting of nonlinear time series using ann, *Future Computing and Informatics Journal* **2**(1): 39–47.

Wang, X., Wu, H., Sun, W. and Zhao, Y. (2019). Towards generating cost-effective test-suite for ethereum smart contract, *2019 IEEE 26th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, IEEE, pp. 549–553.

White, H. (1988). Economic prediction using neural networks: The case of ibm daily stock returns.

Wirth, R. and Hipp, J. (2000). Crisp-dm: Towards a standard process model for data mining, *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, Citeseer, pp. 29–39.

Wu, C.-H., Lu, C.-C., Ma, Y.-F. and Lu, R.-S. (2018). A new forecasting framework for bitcoin price with lstm, *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, IEEE, pp. 168–175.

Yang, D. and Zhang, Q. (2000). Drift-independent volatility estimation based on high, low, open, and close prices, *The Journal of Business* **73**(3): 477–492.