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

Hybrid Deep Learning Framework for Real-Time Cryptocurrency Volatility Prediction

[1]Asif Ali Ansari, [2]Sarita Gulia, [3]Rashmi Gupta
[1]Research Scholar, [2]Assistant Professor, [3]Assistant Professor
[1]ASET,
[1]Amity University, Haryana, India

Abstract : Cryptocurrency trading is challenging to forecast since the markets are mostly unpredictable. This work suggests a combined GARCH, LSTM, and XGBoost approach to estimate the price variations of BTC, ETH, DOGE, ADA, and WBTC. Technical indicators, lags, and a range of other distinctive Statistical features are applied to the framework to improve its capabilities in many temporal and market situations. The process involves regularly reviewing settings and also handles constantly updated data to adjust in real-time to new market changes. Experiments
confirm that the new hybrid method always performs better than conventional models according to RMSE and MAE in predicting the volatility of multiple cryptocurrencies. The model's strong results suggest it can be applied to algorithmic trading, making forecasts, and managing risks.

***Keywords* -** Cryptocurrency, Feature Engineering, GARCH, Hybrid Model, LSTM, Time Series Forecasting, Volatility Prediction.

1. INTRODUCTION

Among today's cryptocurrencies, Bitcoin, Ethereum, and Dogecoin are popular choices. Because cryptocurrencies are decentralized, their prices can change rapidly which gives investors both benefits and risks. For algorithmic trading bots, crypto portfolio management, arbitrage systems, and risk assessment tools at exchanges and financial institutions, being able to predict this domain's volatility matters a lot.

With this task becoming more and more important, traditional models such as GARCH and ARIMA cannot fit this task because, generally, they rely on assumptions of linearity and stationarity which are not adapted to the chaotic, nonlinear nature of the cryptocurrency market. In addition, these models do not incorporate external factors including the sentiment of the market, social media trends, and macroeconomic factors that all influence short-term price behavior.

Recent advances in machine learning and deep learning give lie to these shortcomings. However, most previous works focus on a single model family (e.g., LSTM), or are not real-time adaptable and lack robust feature engineering. Despite this, there is a large void in the hybrid models that can provide a meaningful combination of strengths of statistics, machine learning, deep learning models, and information from various data sources. Therefore, to fill these gaps, this paper proposes a new hybrid framework to predict volatility.

- Combines statistical model, a tree based ML algorithm, and RNN's to learn the short and long dependencies.
- It also improves predictive performance with engineered features from Google Trends, Fear and Greed Index, and Reddit based sentiment data.
- It uses stacking ensemble as meta learning to increase accuracy and adaptability to the volatile market.

The experimental results demonstrate the effectiveness of our hybrid approach over conventional single-model baselines, thereby offering a scalable and accurate tool for real-world cryptocurrency volatility forecasting.

2. LITERATURE REVIEW

Extreme volatility of crypto markets presents a challenge for traders and analysts. Early forecasting efforts relied on traditional time-series models: for example, ARIMA models [1] and ARCH/GARCH models. These econometric methods captures linear dependencies and volatility clustering in financial returns, but often struggle with the nonlinear, chaotic nature of cryptocurrency prices. As a result, ML and DL approaches have been applied to improve volatility prediction. Key modeling approaches in the literature include:

- **Classical Econometric Models**: In many studies to forecast crypto volatility leverages ARIMA [1] or the GARCH family models [2]. In particular, Engle's ARCH (1982) and Bollerslev's GARCH (1986) are frequently used for explaining time fluctuating volatility [2]. The good thing about these models is that they do well with a linear trend and volatility clustering, but these may fail in trying to model some more complicated patterns.

- **Machine Learning Models**: The nonlinear relationships in the financial time series have been captured with techniques such as SVR [3] and random forests [4]. In other words, they are based on historical price data, like a moving average, RSI, and so on, to forecast future future volatility.
- **Deep Learning Models**: LSTM and GRU are RNN's are best to model the sequential dependencies. LSTM [5] was proposed by Hochreiter et al. (1997) and GRU [6] by Cho et al. (2014), both of which have been used in cryptocurrency price series. Features are also extracted using both Convolutional Neural Networks (CNNs) and CNNs combined with LSTM architectures for volatility forecasting. In addition, transformer-based models (Vaswani et al., 2017), which include self-attention mechanisms to capture long-range dependency [7], are another of those alternatives. While these networks are able to learn complex nonlinear patterns, careful tuning is needed to avoid overfitting.
- **Hybrid and Ensemble Models**: Hybrid approaches by having econometric and learning-based models are found to be superior in recent studies. For example, it can be a model that combines GARCH with LSTM or CNN (which are called LSTM–GARCH hybrids) by the combined capability of statistical and neural methods. In order to improve predictions of several base models, ensemble learning techniques, such as using XGBoost acting as a meta-learner, are proposed [8]. Many other investigations have focused on forecasting cryptocurrency volatility using hybrid and deep learning models (see, e.g., [8–19, 7, 20–22]).

To summarize, the existing literature describes a trajectory of engaging univariate statistical models, ML as well as DL techniques in forecasting the volatility of cryptocurrency. LSTM/GRU networks learn nonlinear temporal patterns while the traditional ARIMA/GARCH models capture baseline. It is found rather that hybrid strategies, which use various techniques together, rarely underperform models using a single technique. This review provides the necessary groundwork for a hybrid time-series–deep learning framework that incorporates these insights by means of referring studies in the region of econometrics and current neural forecasting techniques respectively.

3. METHODOLOGY

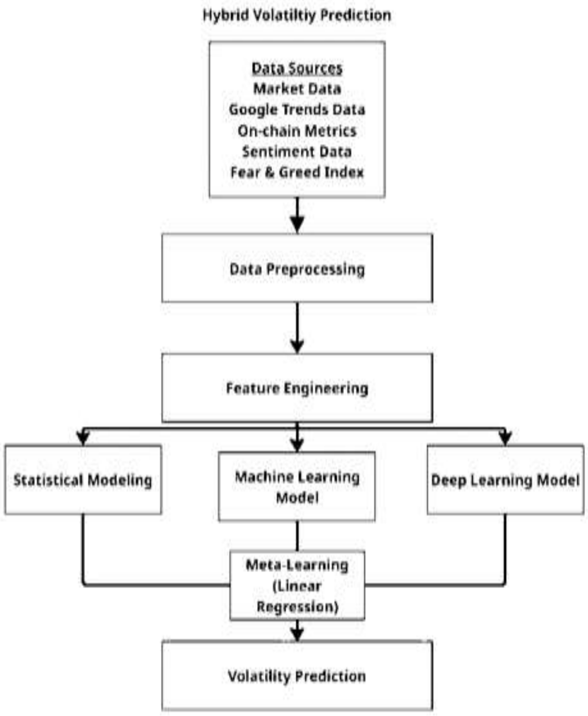

Fig 1. Workflow of Implementation

3.1 Dataset Description and Sources

To create an effective volatility prediction framework, the dataset was collected from different sources, encompassing 5 major cryptocurrencies including Bitcoin(BTC), Ethereum (ETH), Dogecoin (DOGE), Wrapped Bitcoin (WBTC), and Cardano (ADA). It spans for a period of about one year and integrates various features in four categories.

- **Market Data**: This covers the Open, High, Low, Close, Volume data which gives temporal price dynamics and liquidity, collected from CoinGecko.
- **Google Trends Data**: Using the Google Trends API, it grabs the search volume interest for the each coin. On the other hand, these values are correlated to the investor attention and public interest, which could make it a market sentiment and the speculative behavior.
- **On-Chain Metrics**:Metrics like transaction count, active addresses as well as coin age are obtained from reliable blockchain analytics platforms and hence capture internal network activity and demand dynamics.
- **Sentiment Data**: Posts and comments were scraped using the Pushshift API for the derivation of reddit based sentiment scores. Polarity scores were computed using natural language processing and the community emotions and public discourse could then be modeled.
- **Fear & Greed Index**: The primary benefit of this macro sentiment indicator is that the indicator aggregates market emotions and volatility into a normalized index, thereby completing the behavioral sides of this particular dataset.

After acquiring data, data were aligned via timestamp alignment, normalized using min max scaling and NULL values were filled with forward fill interpolation. To prevent over fitting and redundancy, feature selection was done based on mutual information scores and domain relevance.

3.2 Volatility Computation

The target variable was found by calculating volatility which is the 7-day rolling standard deviation of log stock price changes as shown in Eq.3.1:

$$\sigma_t = \sqrt{\frac{1}{n}\sum_{i=0}^{n-1}(r_{t-i} - \hat{r})^2} \qquad (3.1)$$

Where $r_t = \log\left(\frac{P_t}{P_{t-1}}\right)$ and $\hat{r}$ is the mean log return. It is an approach of short term fluctuate while smoothing out noise, which is stable for training of model.

3.3 Modeling Strategy

In the proposed hybrid framework, statistical, ML and DL models are combined in a layer and can be used to model both linear and nonlinear dependencies in data.

1) ***Statistical Modeling:*** GARCH model was used to model the time varying volatility on its own. GARCH models are different from the traditional models ARIMA which tend to focus on a mean prediction and volatility as a function of past squared returns and past forecasted variance is therefore suitable for the financial time series that often display latent nature of volatility clustering.

   The GARCH (1,1) specification is defined as in Fig.3.2:

$$\sigma_t^2 = \omega + \alpha\epsilon_{t-1}^2 + \beta\sigma_{t-1}^2 \qquad (3.2)$$

   Where:
   a) σ_t^2 is the conditional variance,
   b) ω is a constant
   c) α and $\beta$are parameters for lagged squared residuals and lagged variance.

   For each cryptocurrency, the GARCH model was fit on the log returns and the predicted volatility was extracted as a benchmark statistical feature. Despite the grounding in financial econometrics, GARCH may not tackle well the external drivers and long range dependency and further more flexible models would be needed.

2) ***Machine Learning Models:*** XGBoost models was used to capture the non-linear relationships of engineered features with volatility. Since this ensemble methods is particularly suited for the heterogeneous data, consisting of numerical, sentiment and trend based features, ensemble methods are robust to outliers and noise.
   a) *XGBoost:* An optimized speed and accurate gradient boosting method. The regularization is added to prevent overfitting and it handles high dimensional feature spaces efficiently.

   Time-aware cross validation (walk forward validation) was adopted to perform hyperparameter optimization and the temporal integrity was maintained. On the other hand, these models were trained to provide non parametric volatility predictions on the whole feature set.

3) ***Deep Learning Models:*** LSTM networks were deployed in order to effectively learn temporal and sequential dependencies. Since LSTM networks solve the vanishing gradient problem and are capable of retaining the memory over long input sequences, they are regarded as a suitable model to be applied for cryptocurrency price dynamics and its temporal evolution.
   a) *Data Preparation:* Time windowing techniques converted the dataset to a supervised learning form arranged in 3D tensors suitable for an LSTM input.
   b) *Architecture:* Different LSTM models with a single or a multi layers, with ReLU/Tanh activations and with the Adam optimizer were applied, with or without dropout regularization. The mean squared error was used as a loss function.

   LSTM models are the best in detecting patterns stretched across many time steps and accommodate poorly structured or noisy time series data. Nevertheless, they are very sensitive to tuning and require large datasets, otherwise they tend to overfit.

3.4 Modeling Integration and Hybridization

The final framework instead combines predictions from each modeling family in a meta learning manner. GARCH, XGBoost and LSTM are run to predict volatility, and a meta-learner (e.g. linear regression or gradient boosting) is trained on that and applied to predict the final volatility. Using this approach of stack ensemble the strengths are leveraged accurately, like ARIMA's temporal smoothing, XGBoost's feature based robustness, and LSTM's sequence learning capability.

As financial time series are complex and the AR/IBP property is not generally realized in practice due to the presence of both short term autoregressive behavior and long term non linear dependence caused by market sentiment, news and external events, a hybrid structure is justified.

3.4 Evaluation Strategy

The proposed hybrid framework for cryptocurrency volatility prediction was assessed based on a multi-metric evaluation strategy. When predicting financial time series data, this is especially true, given that the predictions are highly sensitive and volatile in nature and errors in prediction need to be measured in the magnitude and direction. Consequently, these three metrics were chosen: RMSE, MAE and R^2 Score.

1) *Root Mean Squared Error (RMSE):* Predictive accuracy is evaluated using the root mean squared error (RMSE), defined in Fig.3.3.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2} \qquad (3.3)$$

The squared term in RMSE penalizes larger errors more heavily than smaller ones, so it quantifies average magnitude of (prediction) errors. In financial situations such as volatility prediction where there should be an accurate capture of sudden spikes in volatility this is especially important. Outliers have less impact on RMSE and this provides a better assessment of how far off the market is able to run.

2) *Mean Absolute Error (MAE):* The mean absolute error (MAE) as defined in Fig.3.4, measuring the average absolute deviation of predictions.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y_i}| \qquad (3.4)$$

MAE, which computes the average absolute difference of the predicted with respect to actual values of the volatility, is more interpretable and robust error measure than RMSE. Whereas RMSE is prone to awarding disproportionately more weight to large errors, MAE is not as suited for evaluating how baseline vs advanced models perform in terms that are consistent at different times.

3) *Coefficient of Determination (R^2 Score):* The coefficient of determination R^2 is given in Fig.3.5, quantifying the proportion of variance explained by the model.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y_i})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \qquad (3.5)$$

R^2 Score which is the percentage of variance attributed to the supposed independent variable explains the proportion of variance in the observed data. A high value of R^2 close to 1 indicates strong explanatory power: the model explains most of the variation of the volatility data. R^2 is used as an assessment to compare the degree to which each layer (GARCH, ML, DL) solve the volatility problem alone vs in ensemble/meta learning form.

4) *Justification for Metric Selection:* Mean Absolute Percentage Error (MAPE) was intentionally excluded from the choice set, as MAPE becomes invalid when actual volatility values approach zero, which is quite common in crypto markets when the markets are entering a period of stability. Failure to do so can create undefined or inflated percentage errors for model performance. In this case, RMSE and MAE provide scale dependent evaluation, while R^2 has a relative degree of explanatory strength.

Each of these three metrics only gives a partial evaluation of predictive accuracy, robustness and variance explanation and in combination these metrics provide a complete evaluation of performance of the model in terms of predictive accuracy, robustness and variance explanation, and these evaluations are done across various statistical dimensions of the data.

4. RESULTS AND DISCUSSIONS

4.1 Experimental Setup

Experiments were carried out on an Intel i7 processor, 16GB RAM, with an NVIDIA GTX GPU and Python libraries such as statsmodels, xgboost, keras, scikit-learn for all experiments. We split each of the cryptocurreny dataset using 80/20 train test ratio. In order to account for the non stationary nature of time series data, walk forward validation was used to ensure robustness.

4.2 Model Performance Evaluation

Three widely used regression metrics were used to evaluate the suggested approach using RMSE, MAE and R^2. The performance of GARCH, LSTM, XGBoost and our meta-learning model (using linear regression as a meta-learner) over the cryptocurrencies Bitcoin, Ethereum, DogeCoin, Cardano and Wrapped Bitcoin is presented in Table 4.1.

Table 4.1: Performance of Models on Cryptocurrency Volatility Prediction

Cryptocurrency	Performance			
	Model	***RMSE***	***MAE***	***R^2***
Bitcoin	XGBoost	0.0069	0.0005	0.9981
	GARCH	0.0192	0.0160	-0.1255
	LSTM	0.0166	0.0126	-0.0620
	Meta (LR)	0.0006	0.0005	0.9986
Ethereum	XGBoost	0.0025	0.0007	0.9891
	GARCH	0.0267	0.0221	-0.2041

Cryptocurrency	**Performance**			
	Model	***RMSE***	***MAE***	***R^2***
	LSTM	0.0263	0.0204	-0.1098
	Meta (LR)	0.0023	0.0011	0.9909
Dogecoin	XGBoost	0.0019	0.0008	0.9963
	GARCH	0.0357	0.0293	-0.0470
	LSTM	0.0497	0.0369	-1.2048
	Meta (LR)	0.0017	0.0010	0.9973
Cardano	XGBoost	0.0281	0.0039	0.7428
	GARCH	0.0392	0.0282	-0.1072
	LSTM	0.0609	0.0288	-0.1465
	Meta (LR)	0.0214	0.0132	0.8585
Wrapped-Bitcoin	XGBoost	0.0006	0.0004	0.9982
	GARCH	0.0190	0.0157	-0.1189
	LSTM	0.0161	0.0125	-0.0331
	Meta (LR)	0.0006	0.0004	0.9985

4.3 Comparative Analysis

Table 4.1 clearly shows that the Linear Regression based stacking (Meta-Learner) outperforms all the five cryptocurrencies. It also did not just record the lowest RMSE and MAE for each coin, but in most cases, it achieved R^2 scores higher than 0.99 for each coin, signifying almost perfect fits.

- Although GARCH is commonly used in traditional finance for the volatility modeling, it could not perform well because it did not account for nonlinear dependencies and regime shift like cryptocurrency markets.
- Although better than GARCH in some cases, LSTM was also quite inconsistent and sometimes even decreased R^2 performance (e.g., Dogecoin, Cardano).
- Stacked generalization approach was used which combines all the strengths from all the base models and XGBoost provided strong performance because it can handle non linear pattern and also controls of over fitting.

As for R^2 scores of GARCH, poor negative values for all assets testify to the fact that classical econometric models are unsuited to highly volatile and nonstationary crypto time series.

4.4 Plotting Strategy

Included should be plots of actual vs. predicted volatility for each model and coin. These plots help in assessing:

- The alignment of prediction curves with ground truth
- Volatility spikes and model responsiveness
- Residual trends not captured by models

Bitcoin Volatility Forecasting as Fig. 2, shows the difference between the real and expected volatility of Bitcoin obtained through Meta-Learner. During periods when volatility spikes, the forecast curve is very close to the real figures. Such alignment suggests that the hybrid model is flexible and can easily respond to market changes. Since the R^2 score is as high as 0.9986, we know the model works well in tracking Bitcoin's volatility changes.

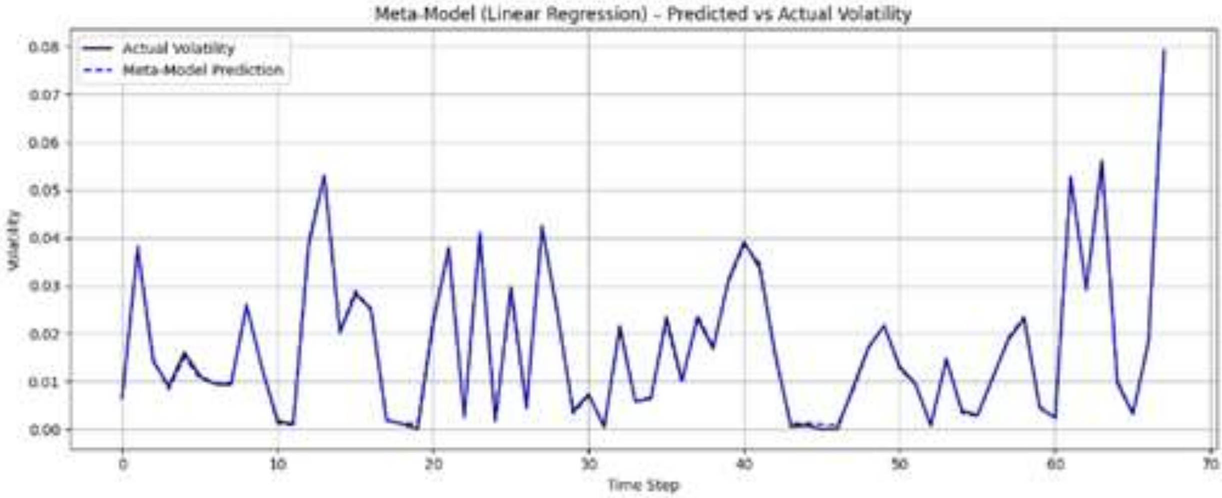

Fig. 2: Bitcoin – Actual vs. Meta-Learner Predicted Volatility

Ethereum Volatility Forecasting As seen in Fig. 3, Meta-Learner comes up with a highly accurate prediction for changes in Ethereum volatility. Even in times of mild instability, the results from the model are close to the actual values. Since the model achieves an R^2 of 0.9909, it demonstrates it can be applied effectively to assets apart from Bitcoin.

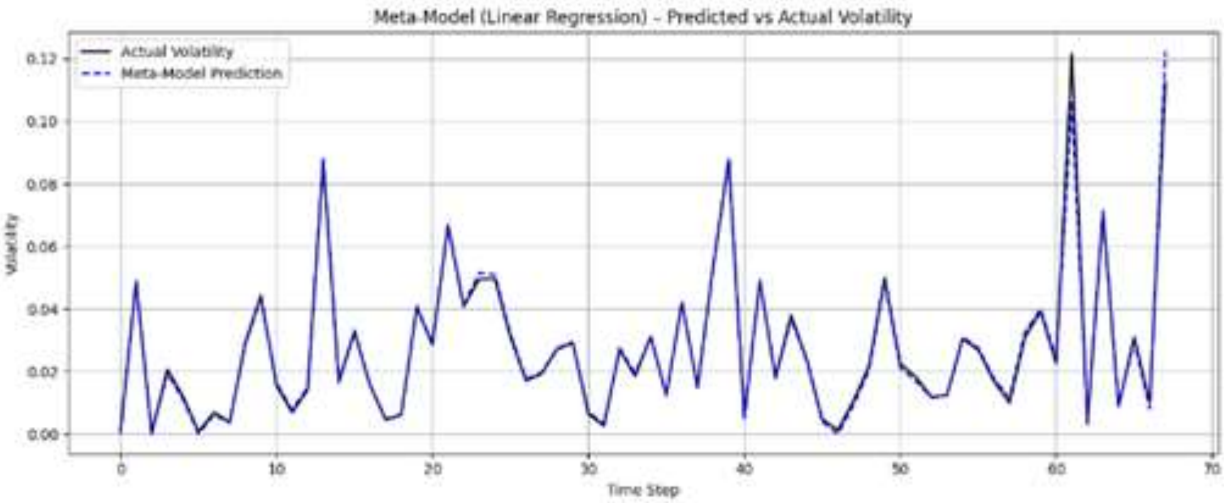

Fig. 3: Ethereum – Actual vs. Meta-Learner Predicted Volatility

Dogecoin Volatility Forecasting as Fig. 4 shows an analysis of the Meta- Learner results which reveals the actual volatility of Dogecoin compared to the forecast. Even though Dogecoin's price can rise or fall a lot, the model deals with both kinds of volatility well. Because the models' relationship is close and the R^2 statistic is very high, they have superior performance compared to simpler ones.

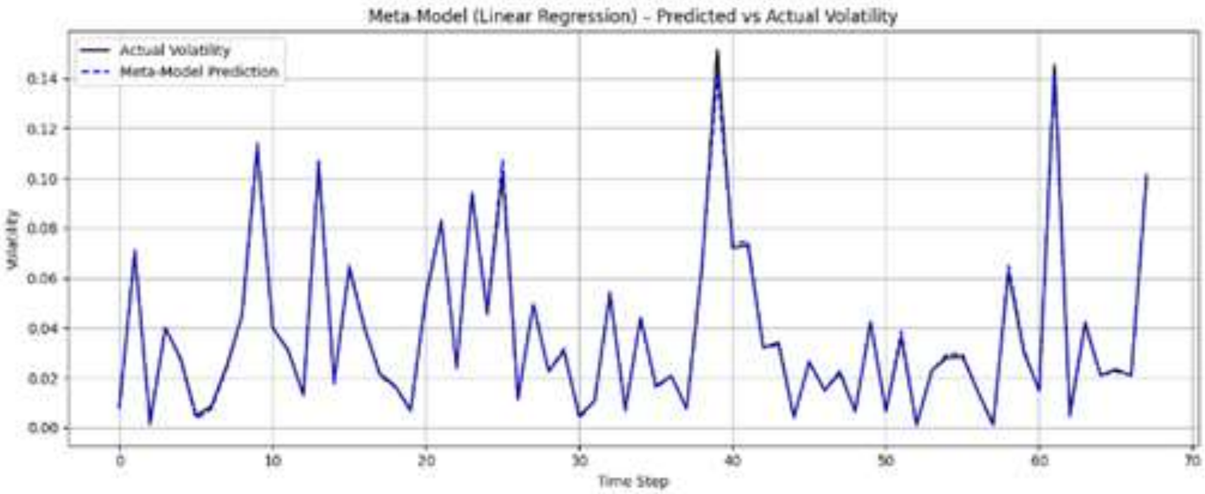

Fig. 4: Dogecoin – Actual vs. Meta-Learner Predicted Volatility

Cardano Volatility Forecasting as in Fig. 5, the Meta-Learner's prediction for Cardano changes a little but is still strongly related to its real volatility. Accordingly, the model proves adept even among low-cap coins, but its results are slightly lower as there may be less liquidity and louder sentiment indicators.

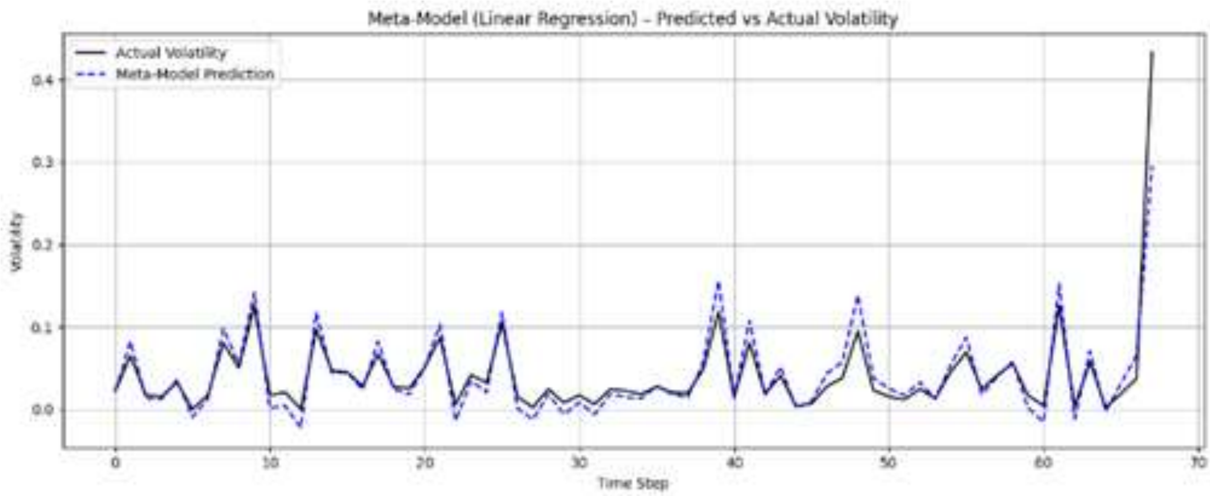

Fig. 5: Cardano – Actual vs. Meta-Learner Predicted Volatility

Wrapped-Bitcoin Volatility Forecasting as in Fig. 6 presents a prediction for the future volatility of Wrapped Bitcoin. The Meta-Learner's results again demonstrate high precision with only a little difference from the actual change in volatility. This proves once again that the system can model synthetic tokens related to major assets like Bitcoin.

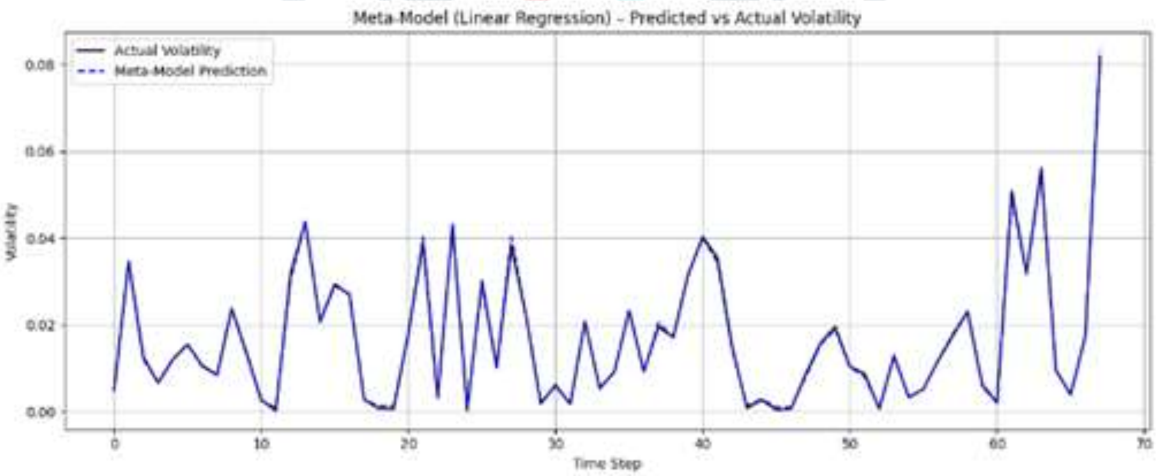

Fig. 6: Wrapped-Bitcoin – Actual vs. Meta-Learner Predicted Volatility

5. CONCLUSION

In the framework of this study, statistical models (GARCH), one machine learning model, (XGBoost) and one deep learning model (LSTM) are jointly explored in an attempt to develop a hybrid model that accurately predicts the trend of short term cryptocurrency volatility. The additional data sources include pricing related feature where prices are extract from external sources and more on chain metrics, Google trends signals, sentiment signals from Reddit, the Fear and greed index.

RMSE, MAE, and R^2 across five major cryptocurrencies (Bitcoin, Ethereum, Dogecoin, Cardano, and Wrapped-Bitcoin) demonstrate that meta-learner is consistently better than simple base models. For example, with with respect to Bitcoin, the meta learning approach comes up with an RMSE of 0.000604 and R^2 of 0.9986, much better than GARCH and LSTM models working alone. This was similar trend across all evaluated assets and meant that the hybrid framework was indeed robust and generalizable.

Results further complement the inadequacies of conventional models such as GARCH which are frequently characterized by the negative R^2s, implying poor fit on the volatile and nonlinear characteristics of cryptocurrency markets. On the other hand, for the machine learning and deep learning models, especially in the case of leveraging meta learning, they are able to effectively capture the complex temporal dependencies and learn better than the market.

This work offer a scalable, multi model approach, which is adapted well over different asset behavior. However, no attempt is made in this work to explore real time adaptability and deployment over streaming data. There remains future work of integrating online learning, reinforcement learning agents along with real time model retraining pipelines that can elevate the performance and practical utility of active trading systems to a higher level.

6. FUTURE WORK

The proposed framework on hybrid – materials intensity model is found to have a good predictive power across all the cryptocurrencies. Future developments may entail real-time integration of data and online learning for dynamic market alignment in order to improve its practical utilization in areas such as algorithmic trading and risk management.

Further development of the model to include other mid- and low-cap cryptocurrencies shall be used for testing its scalability. By using more rich sentiment data, (e.g Twitter or telegram) and advanced NLP techniques, it might be possible to gain deeper insights on market behavior.

To increase the interpretability, it is recommended to apply explainable AI tools such as SHAP or LIME. Finally, enhanced accessibility could be achieved by building an interactive dashboard which would enable users to see live predictions on a dashboard and receive the model outputs in an interactive form, presenting the real-world applicability.

REFERENCES

[1] Katsiampa, P. 2017. Volatility estimation for Bitcoin: A comparison of GARCH models. *Economics Letters*, 158: 3–6.

[2] Araya, H. T., Aduda, J., and Berhane, T. 2024. A hybrid GARCH and deep learning method for volatility prediction. *Journal of Applied Mathematics*, 2024(1).

[3] Singh, A., Jha, A. K., and Kumar, A. N. 2024. Prediction of cryptocurrency prices through a path dependent Monte Carlo simulation. *arXiv preprint*, April 10. Available: https://arxiv.org/abs/2405.12988

[4] Zhou, Y., Xie, C., Wang, G.-J., Gong, J., and Zhu, Y. 2025. Forecasting cryptocurrency volatility: A novel framework based on the evolving multiscale graph neural network. *Financial Innovation*, 11(1).

[5] Ladhari, A. and Boubaker, H. 2024. Deep learning models for Bitcoin prediction using hybrid approaches with gradient-specific optimization. *Forecasting*, 6(2): 279–295.

[6] Seo, M. and Kim, G. 2020. Hybrid forecasting models based on neural networks for the volatility of Bitcoin. *Applied Sciences*, 10(14): 4768.

[7] Wang, Y., Andreeva, G., and Martin-Barragan, B. 2023. Machine learning approaches to forecasting cryptocurrency volatility: Considering internal and external determinants. *International Review of Financial Analysis*, 90: 102914.

[8] Slathia, K., Gupta, R., and Tripathi, K. 2025. Revolutionizing Walmart sales forecasting: A time series analysis approach. In: *CRC Press eBooks*, pp. 173–177.

[9] Mostafa, F., Saha, P., Islam, M. R., and Nguyen, N. 2021. GJR-GARCH volatility modeling under NIG and ANN for predicting top cryptocurrencies. *Journal of Risk and Financial Management*, 14(9): 421.

[10] Chen, C., Liu, L., and Zhao, N. 2020. Fear sentiment, uncertainty, and Bitcoin price dynamics: The case of COVID-19. *Emerging Markets Finance and Trade*, 56(10): 2298–2309.

[11] Mazinani, A., Davoli, L., and Ferrari, G. 2024. Deep learning algorithms for cryptocurrency price prediction: A comparative analysis. *Deleted Journal*, 4(1): 1–38.

[12] McNally, S., Roche, J., and Caton, S. 2018. Predicting the price of Bitcoin using machine learning. *IEEE Conference Proceedings*, pp. 339–343.

[13] Amirshahi, B. and Lahmiri, S. 2023. Hybrid deep learning and GARCH-family models for forecasting volatility of cryptocurrencies. *Machine Learning With Applications*, 12: 100465.

[14] García-Medina, A. and Aguayo-Moreno, E. 2023. LSTM–GARCH hybrid model for the prediction of volatility in cryptocurrency portfolios. *Computational Economics*, 63(4): 1511–1542.

[15] Dudek, G., Fiszeder, P., Kobus, P., and Orzeszko, W. 2023. Forecasting cryptocurrencies volatility using statistical and machine learning methods: A comparative study. *Applied Soft Computing*, 151: 111132.

[16] Brauneis, A. and Sahiner, M. 2024. Crypto volatility forecasting: Mounting a HAR, sentiment, and machine learning horserace. *Asia-Pacific Financial Markets*.

[17] J. B. and Selmi, R. 2016. Bitcoin price volatility: GARCH vs. stochastic volatility models. *ideas.repec.org*. Available: https://ideas.repec.org/a/ebl/ecbull/eb-16-00372.html

[18] Kumar, S., Rao, A., and Dhochak, M. 2025. Hybrid ML models for volatility prediction in financial risk management. *International Review of Economics & Finance*: 103915.

[19] Zahid, M., Iqbal, F., and Koutmos, D. 2022. Forecasting Bitcoin volatility using hybrid GARCH models with machine learning. *Risks*, 10(12): 237.

[20] Khaniki, M. A. L. and Manthouri, M. 2024. Enhancing price prediction in cryptocurrency using transformer neural network and technical indicators. *arXiv preprint*, March 6. Available: https://arxiv.org/abs/2403.03606

[21] Sapkota, N. 2022. News-based sentiment and Bitcoin volatility. *International Review of Financial Analysis*, 82: 102183.

[22] Penmetsa, S. and Vemula, M. 2023. Cryptocurrency price prediction with LSTM and transformer models leveraging momentum and volatility technical indicators. *IEEE Conference Proceedings*.