

Out of the Box: Deep Learning for Volatility Forecasting

A THESIS

Presented to

The Faculty of the Department of Economics and Business

The Colorado College

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Arts

By:

Omar Castro-Frederick

May 2023

Out of the Box: Deep Learning for Volatility Forecasting

Omar M. Castro-Frederick

May 2023

Mathematical-Economics

Abstract

We inspect how effective out of the box multivariate time series Transformer models are in forecasting a volatility index and how viable they are for future general use for volatility index forecasting by non-machine learning engineers. We analyze the performance of a traditional ARIMA-GARCH econometric model and compare it with the performance of an out of the box Light Gradient Boosting Machine and an out of the box Multivariate Time Series Transformer machine learning model. We implement our machine learning models with no hyperparameter tuning and little feature generation. Despite the low level of model refinement, the multivariate time series transformer outperformed the GARCH model and demonstrated a greater ability to predict when there would be changes in volatility. This is likely due to the use of positional encoding that allows the multivariate time series transformer to learn the relationship between nearby datapoints. We propose the development of more accessible time series transformers for use in volatility forecasting.

Key Terms

ACRONYMS:

AR-GARCH: The ARIMA-GARCH model is a hybridized econometric model that uses the linear ARIMA model and the non-linear GARCH model.

LGBM: The Light Gradient Boosting Machine is a tree based machine learning algorithm.

MTST: The Multivariate Time Series Transformer is a variant of the Transformer architecture.

KEYWORDS:

Transformers: Transformers are a type of advanced neural network that has traditionally used an encoder-decoder architecture with a self-attention mechanism. These models have become widely used for language modeling tasks. It is worth noting that time series adaptations of these architectures have generally stopped using the decoder.

Multivariate Time Series Transformer: The Multivariate Time Series Transformer is a Transformer variant without an encoder that uses relative positional encodings and batch normalization.

JEL CODES: C01, C32, C45, C53, C58, C87, C88, G18

ON MY HONOR, I HAVE NEITHER GIVEN NOR RECEIVED
UNAUTHORIZED AID ON THIS THESIS

Omar M. Castro-Frederick

Signature

Table of Contents

Abstract	II
Key Terms	III
List of Tables	VI
List of Figures	VI
Acknowledgments	VII
1. Introduction	1
3. Methodology	4
3.1. Data	4
3.2. AR-GARCH	4
3.3. LGBM	5
3.4. Transformer	6
4. Results and Analysis	9
5. Conclusion	11
References	12

List of Tables

Table 1: Shows the correlation matrix between the variables we generate for making predictions. Although all these variables are off by several steps it demonstrates that even a step difference of ten has a non-negligible linear predictive ability.

Table 2: Shows the resulting metrics from the predictions generated by the GARCH, LGBM, and MTST models.

List of Figures

Figure 1: Demonstrates the difference between level wise and leaf-wise growth.

Figure 2: Is a diagram of general architecture of a time series transformer created by Ramos-Pérez, E., Alonso-González, P. J., & Núñez-Velázquez, J. J. (2021)

Figure 3: Are four graphs that show the volatility index predictions, volatility index error, volatility index variance, and volatility mean squared error.

Acknowledgments

I would like to acknowledge my advisor Professor Michelan Wilson for guiding me through the process of writing this thesis and for encouraging and allowing me to move forward with a topic that combines my experience with Computer Science in Economics.

1. Introduction

In order to make informed financial decisions about when to invest investors have either implicitly or explicitly taken stock market volatility into account when assessing risk. Since Robert Engle's paper on the Autoregressive Conditional Heteroscedastic models in 1982, Econometrics have been used to forecast volatility in the stock market. (Engle, 1982) Econometric models have historically been used for univariate and multivariate predictions of stock market volatility, but with the rise of machine learning(ML) models there have come new and often more accurate methods at predicting market volatility. There are a number of prominent econometric models used for forecasting stock market volatility using univariate data but two of the most prominent ones are the GARCH and ARIMA(AR) models. The AR-GARCH model on the other hand, has been demonstrated to be able to make use of ARIMAs linear predictions and GARCHs non-linear predictions creating a more effective model for stock market volatility predictions. (Chand et al., 2012) Since the AR-GARCH model has been demonstrated to be effective in multivariate stock volatility index forecasting we will be using it as our econometric baseline. (Schreiber, 2009; Russel et al., 2020)

While econometric models have been viable for forecasting volatility, the great strides made in ML and neural networks have made models with deep learning a more viable option for predicting stock market volatility.

With the rise in use of Transformers for language modeling for sequence to sequence tasks, there has been an increased interest in making use of Transformer architecture for other tasks given how effective they have been demonstrated to be in deep learning tasks. One such area is time series forecasting and classification, to make it possible for a transformer to handle time series data, the Multivariate Time Series Transformer(MTST) does not make use of the decoder used for sequence generation found in the original Transformer. (Zerveas et al., 2021)

Although the MTST has a simpler architecture than its language model counterpart it has shown to be highly effective in making time series predictions in both regression and classification tasks, outperforming other state of the art prediction algorithms like: Rocket, SVR, XGBoost, and LSTMs.(Zerveas, 2021)

It has been reported that there are predictable patterns in stock volatility, even so predictable patterns are self-destructive as investors take advantage of these patterns adding more unpredictability.(Marquering & Verbeek, 2023) Due to the relationship between learning and decreasing levels of predictability, those who are able to create more advanced predictive models will generally have an edge over those who use more rudimentary models.

Due to the MTSTs ability to make use of positional encodings, it is able to make predictions based on sequential data. Since the Transformer uses deep learning(the use of several layers of cells) for making predictions it is able to identify and develop

more complicated pattern identification. (LeCun et al., 2015) Another feature Transformers employ is the self attention mechanism which allows the model to develop relationships between different positions within the sequence enabling transformers to scale more effectively when used in conjunction with multi headed attention. (Vaswani et al., 2017) Time series Transformers have already been demonstrated to be highly effective at predicting stock market volatility. In Ramos-Pérez, E., Alonso-González, P. J., and Núñez-Velázquez, J. J., 2021 paper “Multi-Transformer: A New Neural Network-Based Architecture for Forecasting S&P Volatility” it was demonstrated that time series transformers can be highly effective at predicting stock volatility, although this model may have some shortcomings when compared to the MTST. Although the Multi-Transformer models were demonstrated to be highly effective in predicting stock volatility, there are two distinct differences between the Multi-Transformer and the MTST that may give the edge to the MTST model over the Multi-Transformer. The MTST uses batch-normalization instead of layer normalization which mitigates the effect that outliers have on a model and it also uses trainable positional encodings instead of absolute positional encodings which were demonstrated to perform better than the absolute positional encodings used in the Multi-Transformer. It is also worth noting that the MTST was designed for general predictive purposes and has been successful with forecasting energy consumption, moisture levels, and levels of air pollution, while the Multi-Transformer was specifically designed for volatility forecasting. (Zerveas et al., 2021)

Although the use of advanced deep learning machine learning models is fairly new, the use of machine learning in stock volatility predictions has existed since Donaldson and Kamstras hybridized feed-forward neural network that demonstrated an ability to be effective on stock volatility forecasting in 1997. Although their ANN-GARCH model did not outperform a GARCH model, the use of deep learning allowed their feed-forward neural network to capture the pattern of increased volatility after having had negative returns as well as the effects of asymmetrical conditional volatility, both patterns that the GARCH model failed to capture.

In recent years, due to the improvements in computer technology more advanced models have become viable for stock volatility predictions like the LSTM-GARCH, (Kim and Won, 2018), LSTM(PCA), LSTM(AE) (Chen & Hu, 2022) and the Multi-Transformer(Ramos-Pérez, 2021) all of which have state of the art results. Even so, these models are resource intensive to complete and are not widely available.

Although deep learning models are quite effective they also require extensive hyperparameter tuning, feature creation, hybridizing, and model modifications in order to perform effectively creating a large barrier to entry for beginners and individuals with limited programming ability.

Economists in particular would benefit from expanding the scope of the tools they use, machine learning provides improvements over traditional econometric models in

regression, classification, unsupervised learning, and matrix completion tasks. (Athey & Imbens, 2019)

We will be testing whether more advanced models like the MTST are viable options as an additional tool for stock volatility forecasting. For this reason we will be training an out of the box model and evaluating it without further refining the model. (Hossain & Douglas, 2021) With an effective out of the box model, smaller companies and individuals with limited resources would be able to make effective volatility predictions without the limits imposed by the costs of refining a model. The lack of availability of deep learning models means use is mostly limited to machine learning engineers. Deep learning models that do not require an extensive training process would provide individuals without the resources to develop a customized deep learning model to be able to develop and use a deep learning model.

There has been extensive research into the use of machine learning as a better alternative for predicting stock market volatility. Even so, there has been no use of the Time Series Transformer for predicting stock market volatility. For the previously aforementioned reasons, the out of the box MTST model will be used for forecasting stock volatility and the AR-GARCH model will be used as the baseline. Additionally, the Light Gradient Boosting Machine(LGBM), a tree based algorithm, will be used as an out of the box regression tree baseline since it is more generalizable than other regression tree algorithms.

2. Methodology

Data

The data that will be used is the S&P 100 volatility index from the FRED database. The previous volatility for the previous day, the previous 2nd day, the previous 5th day, and the previous 10th day is generated for training the AR-GARCH, LSTM, and the MTST model.

	Vol. Index	Delta 1 day	Delta 2 days	Delta 5 days	Delta 10 days
Vol. Index	1	0.9666176759	0.9430780191	0.8993745434	0.8347161656
Delta 1 day	0.9666176759	1	0.9666203402	0.9137265033	0.845635823
Delta 2 days	0.9430780191	0.9666203402	1	0.9288659808	0.8541254824
Delta 5 days	0.8993745434	0.9137265033	0.9288659808	1	0.8994223134
Delta 10 days	0.8347161656	0.845635823	0.8541254824	0.8994223134	1

Table 1. Correlation Matrix of variables used.

We will be using these specified previous days to provide and emphasize information about both the recent past, as well as previous periods. This information is particularly useful for the AR-GARCH and LSTM models since they do not have the ability to make relationships based on positional encodings.

ARIMA-GARCH

The Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model, introduced by Robert Engle in 1982, is a powerful time series model used in econometrics and financial analysis to study and forecast volatility in financial and economic data. Its significance is particularly pronounced in financial time series analysis due to its ability to account for the clustering or persistence of volatility, where periods of high volatility tend to follow one another.

The GARCH model consists of two primary components: an autoregressive component that models the conditional mean of the time series. This component implies that the value of the time series at time t is influenced by the previous $t-1$ values of the series. And a conditional volatility component which addresses the conditional variance or volatility within the time series.

The GARCH model provides a framework for capturing both short-term and long-term memory effects in volatility by considering the historical conditional variances and squared error terms' influence on current volatility but there are several assumptions that limit how effective the GARCH model can be. GARCH models typically assume that the time series is stationary, which means that its statistical properties do not change over time. GARCH models assume that positive and negative shocks have symmetrical effects on volatility. In reality, many financial time

series exhibit volatility clustering, where large negative shocks tend to have a stronger impact on volatility than large positive shocks. (Engle, 1982)

The ARIMA model on the other hand is a linear model that uses previous values as variables to predict future values using an optimization routine. This model is most appropriate to use when your variables have high correlation coefficients, a requirement that table 1 shows is satisfied. (Shumway et al., 2017)

When the non-linear model GARCH model is combined with the linear ARIMA model, a model that uses moving averages to make predictions, the combined model captures conditional and unconditional variances. (Box et al., 2015; Zhou et al., 2005)

For the AR-GARCH model we first fit our training data to an ARIMA model, once that model is fit on our data we fit a GARCH model on the residuals of the ARIMA model. Then, once our models are trained we make predictions by passing our validation set to the ARIMA model, again we pass the residuals to the GARCH model. The results of the ARIMA model and the GARCH model are then added to provide us with a prediction.

LGBM

The LGBM, introduced by Microsoft in 2016, is a powerful forecasting model used in forecast volatility in a wide variety of fields including financial and economic data which will be used as the regression tree benchmark. The LGBM has been quite effective with financial time series analysis due to its ability to be trained effectively on a variety of different datasets and make non-linear predictions. (Tutica et al., 2022; Aziz et al., 2022)

The LGBM model is composed of many sub components. The LGBM model uses the GOSS algorithm that uses subsets of the data to make predictions and then uses a greedy algorithm to make an estimation of the most effective model. (Ke et al., 2017)

The decision trees LGBMs use regression trees that then split the data into subsets and use an average of the regression tree outcomes to generate the final output. This mechanism allows the decision tree to use the different features in the data to effectively help come to the final output.

The LGBM is fit to the data then the predictions are generated. In the context of an LGBM trees are generated on subsets of the data while optimizing for the root mean squared error (RMSE) using gradient optimization, a process by which the gradient is minimized in order to reduce the loss function. (Friedman, 2001) Which can be expressed as the minimization of following function:

$$L^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t)$$

This method of gradient optimization allows the decision trees to fit the dataset more closely. (Fafalios, 2020)

Unlike other decision tree based machine learning models, LGBMs use leaf-wise growth instead of level-wise, which is the key to making the LGBM faster than most other decision tree based algorithms. (Ke, 2017)

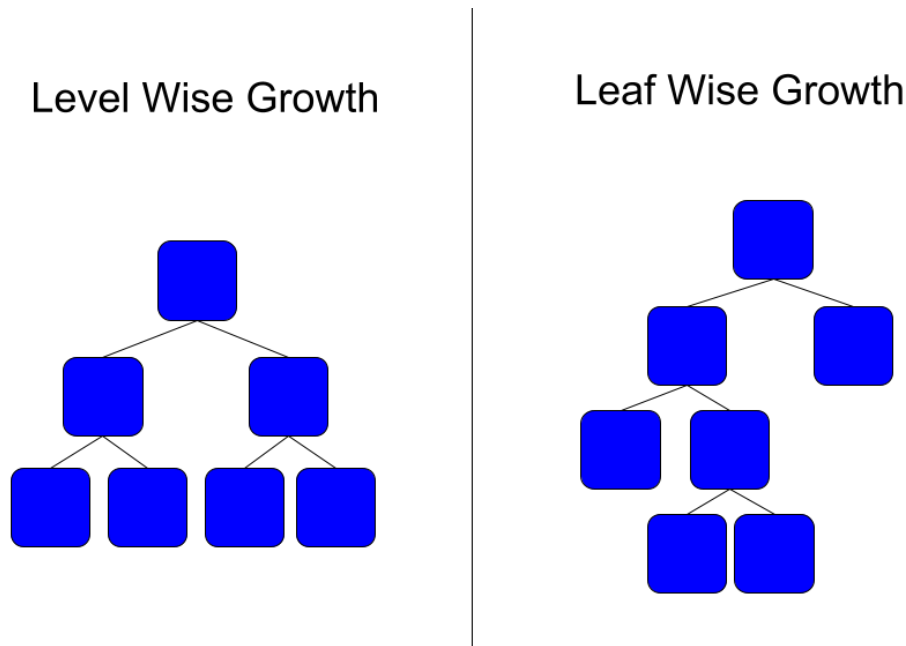


Figure 1. Demonstration of level wise versus leaf wise growth.

The emphasis on leafwise growth means that the model will prioritize improving the model at every step, which may result in the model missing the optimal solution but it does make the model more generalizable than models like XGBoost that require extensive hyperparameter tuning and it also makes the model less prone to overfitting.

Fortunately this process makes the out of the box LGBM model the easiest to train since we simply have to fit the model on our data and then we can make predictions.

Transformer

The MTST is a type of neural network that takes a vanilla Transformer and removes the decoder to allow for the processing of two dimensional data. Vanilla Transformers are unable to make use of time series data since they depend on using three dimensional inputs.

The MTST takes in a generates a vector from the inputs using the encoder and normalizes it for each dimension before projecting that vector into a vector of the

same dimensional space as the transformers sequence dimension. Through the following equation:

$$u_t = W_p x_t + b_p$$

where u_t are the models input vectors, x_t is the feature vectors with the positional encoding added, W_p is a the learnable parameters, and b_p is also learnable parameters accounting for error. (Zerveas et al., 2021) These learnable parameters make up the multi headed self-attention component of the transformer allowing it to modify its outputs based on its other inputs generating learned relationships in between them.

For the regression aspect of the MTST it uses the equation:

$$\hat{y} = W_o \bar{z} + b_o$$

where \hat{y} is the predicted value, W_o and b_o are linear layers with parameters and \bar{z} is the final vector representation. This aspect of the MTST is the feed forward layer which takes the self-attention output and generates the models output.

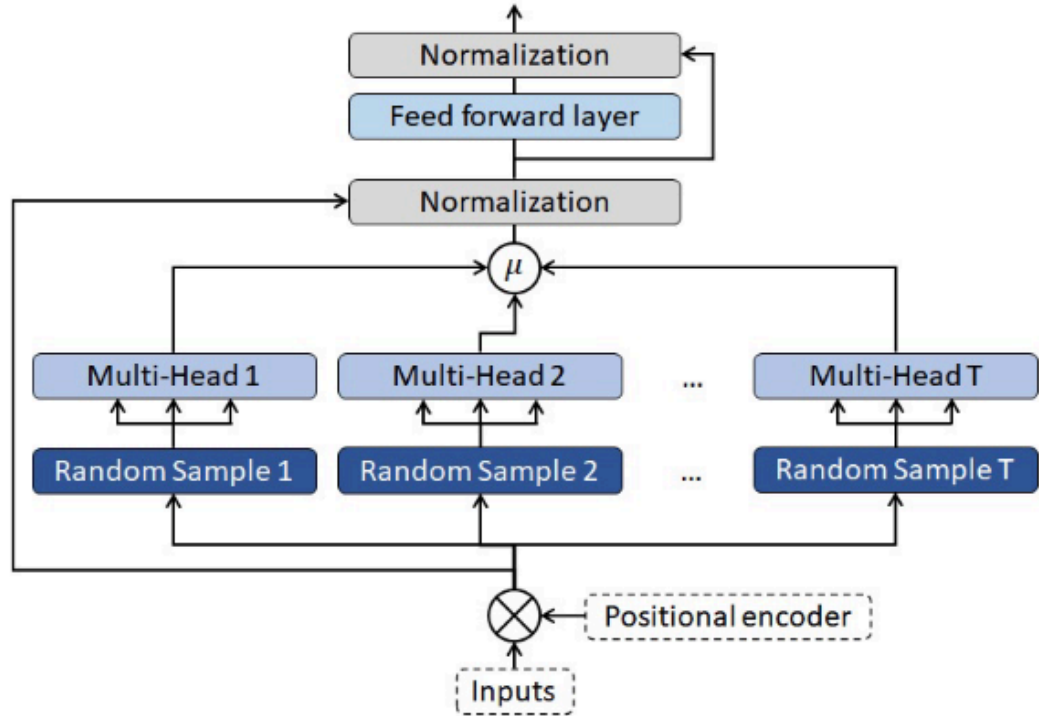


Figure 2. MTST architecture. Ramos-Pérez, E., Alonso-González, P. J., & Núñez-Velázquez, J. J. (2021)

The first step in training the model after having selected for the optimal columns is to fit the model to the data. Fitting the model works by iteratively performing forward propagation, a process in which the input data is passed layer by layer and each neuron applies a transformation allowing the model to make non-linear predictions. (Cai, 2023) In order to get adequate results the MTST went through 2000 fitting

cycles. Then the forward propagation step has been completed and the results are used to compute the loss of every prediction based on the given loss function. The loss is then used in the backwards propagation step in which the calculated loss is used to modify the weights in the model's layers or the learnable parameters. (Wu, 2021) The model with the lowest loss is then saved for making predictions.

3. Results and Analysis

The mean-squared error(MSE), root mean-squared error (RMSE), mean average error (MAE), the coefficient of determination R^2 , and the adjusted coefficient of determination (Adj. R^2) are used as the metrics for evaluating how effective the models are in forecasting the stock volatility index.

	MSE	RMSE	MAE	R^2	Adj. R^2
MTST	4.172	2.043	1.360	0.854	0.865
LGBM	4.588	2.142	1.463	0.840	0.851
AR-GARCH	4.199	2.049	1.384	0.854	0.865

Table 2. The results of running the model on the validation set.

As we can see in table 2 the out of the box MTST model out performed the AR-GARCH model in MAE, MSE, and RMSE and matched the AR-GARCH model in R^2 and adj. R^2 . On the other hand, the LGBM failed to match or outperform the AR-GARCH or MTST in any metrics.

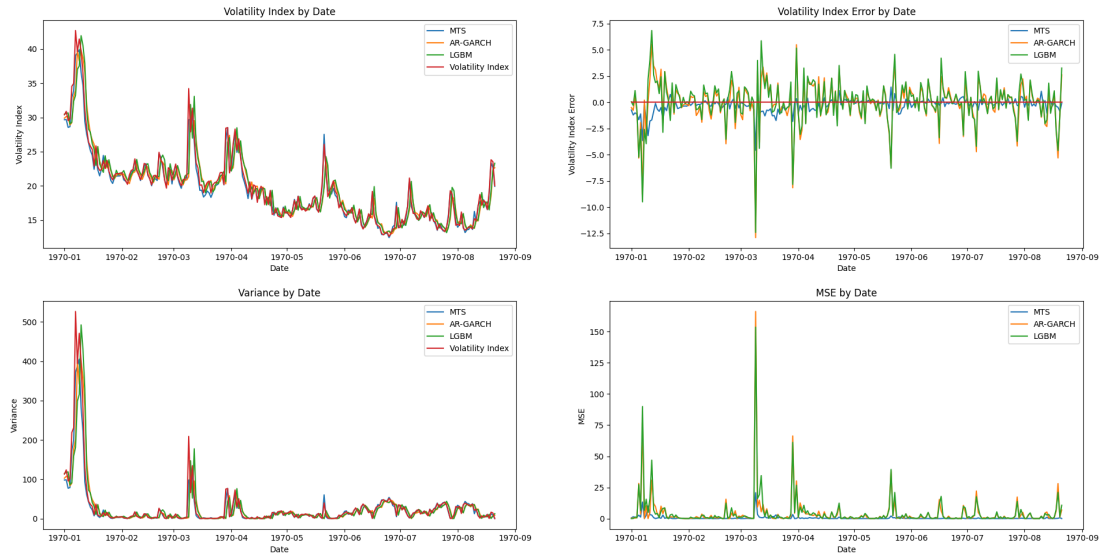


Fig 3. Graph (0,0) shows the volatility index and the predictions of the models. Graph(0,1) shows the error rate for each model at their given predictions. Graph (1,0) shows the squared error for each prediction. Graph(1,1) shows the variance for each given prediction.

As we can see in figure 3 during periods of rapid changes in volatility the LGBM and AR-GARCH models tend to underestimate the change in volatility and then overcompensate by over predicting the volatility creating a lag in the predictions of the LGBM and AR-GARCH models. The MTST model appears to more precisely predict when the stock volatility index will increase in volatility but either under or overestimate the increase in volatility. During the periods of moderate volatility the

MTST model appears to have a very low error rate and all models appear to perform comparably well during periods of low volatility.

The MTST's ability to predict the rapid changes in volatility can be attributed to its ability to leverage positional encodings and the subsequent relational learning, a feature both the AR-GARCH and LGBM models lack.

The MTST's performance can also be attributed in part to using a large proportion of the data to train the model. This likely indicates that the model would perform better with the use of a larger sample size. Additionally, given the use of multi headed self attention the MTST would scale more easily than the AR-GARCH and LGBM models since the model generation would not scale one to one in space and complexity.

4. Conclusion

In this article, classical machine learning methods, econometric methods, and deep machine learning methods were used as a way of determining if an out of the box multivariate time series transformer would be more effective at predicting the stock volatility index. Since closely and consistently predicting the stock volatility index is more beneficial for forecasting the stock volatility index the models will be evaluated on the RMSE of each model. The MTST model performed better than the AR-GARCH and LGBM models even without the use of a hybridized model that have been generally used by other state of the art stock volatility index models. (Hochreiter & Schmidhuber, 1997; Kim & Won, 2018; Ramos-Pérez et al., 2021)

Due to the time dependent nature of the data the LGBM was at an inherent disadvantage but due to its previous success in volatility forecasting the lackluster performance indicates that more preprocessing and feature selection is required to make the LSTM a viable.

It is worth noting that although the MTST model performed better than the LGBM and AR-GARCH models it required a much longer period to train compared to the other two models. Additionally, given how much effort generally goes into refining Transformer models it may be more difficult to generalize the MTST model compared to AR-GARCH and the LGBM.

Despite the limitation of the MTST the success of the out of the box MTST suggests the MTST may be viable as a general tool for future stock volatility forecasting without requiring the extensive resources for model refinement. Since current models are highly technical to work with and often lack documentation, creating software packages that make MTST more readily available for use as an unboxed model would make deep learning for stock volatility index predictions a viable option for individuals who are not machine learning engineers.

References

- Chand, S., Kamal, S., & Ali, I. (2012). Modeling and volatility analysis of share prices using ARCH and GARCH models. *World Applied Sciences Journal*, 19(1), 77-82.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.
- Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica: Journal of the econometric society*, 987-1007.
- Assaf, O., Di Fatta, G., & Nicosia, G. (2021, October). Multivariate LSTM for stock market volatility prediction. In *International Conference on Machine Learning, Optimization, and Data Science* (pp. 531-544). Cham: Springer International Publishing.
- Hutchins, D., Schlag, I., Wu, Y., Dyer, E., & Neyshabur, B. (2022). Block-recurrent transformers. *Advances in Neural Information Processing Systems*, 35, 33248-33261.
- Wu, N., Green, B., Ben, X., & O'Banion, S. (2020). Deep transformer models for time series forecasting: The influenza prevalence case. *arXiv preprint arXiv:2001.08317*.
- Kim, H. Y., & Won, C. H. (2018). Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models. *Expert Systems with Applications*, 103, 25-37.
- Marquering, W., & Verbeek, M. (2004). The economic value of predicting stock index returns and volatility. *Journal of Financial and Quantitative Analysis*, 39(2), 407-429.
- Donaldson, R. G., & Kamstra, M. (1997). An artificial neural network-GARCH model for international stock return volatility. *Journal of Empirical Finance*, 4(1), 17-46.
- Ramos-Pérez, E., Alonso-González, P. J., & Núñez-Velázquez, J. J. (2021). Multi-transformer: A new neural network-based architecture for forecasting S&P volatility. *Mathematics*, 9(15), 1794.
- Zerveas, G., Jayaraman, S., Patel, D., Bhamidipaty, A., & Eickhoff, C. (2021, August). A transformer-based framework for multivariate time series representation learning. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining* (pp. 2114-2124).

- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.
- Tutica, L., Vineel, K. S. K., & Mallick, P. K. (2022). LGBM-Based Payment Date Prediction for Effective Financial Statement Management. In *Cognitive Informatics and Soft Computing: Proceeding of CISC 2021* (pp. 445-455). Singapore: Springer Nature Singapore.
- Aziz, R. M., Baluch, M. F., Patel, S., & Ganie, A. H. (2022). LGBM: a machine learning approach for Ethereum fraud detection. *International Journal of Information Technology*, 14(7), 3321-3331.
- Jung, J. O., Crnovrsanin, N., Wirsik, N. M., Nienhüser, H., Peters, L., Popp, F., ... & Schmidt, T. (2023). Machine learning for optimized individual survival prediction in resectable upper gastrointestinal cancer. *Journal of Cancer Research and Clinical Oncology*, 149(5), 1691-1702.
- Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.
- Wu, D. Y., Lin, D., Chen, V., & Chen, H. H. (2021, October). Associated learning: an alternative to end-to-end backpropagation that works on cnn, rnn, and transformer. In *International Conference on Learning Representations*.
- Cai, G., Zhu, Y., Wu, Y., Jiang, X., Ye, J., & Yang, D. (2023). A multimodal transformer to fuse images and metadata for skin disease classification. *The Visual Computer*, 39(7), 2781-2793.
- Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.
- Hossain, R., and Douglas Timmer. "Machine learning model optimization with hyper parameter tuning approach." *Glob. J. Comput. Sci. Technol. D Neural Artif. Intell* 21.2 (2021).
- Zhou, B., He, D., Sun, Z., & Ng, W. H. (2005, July). Network traffic modeling and prediction with ARIMA/GARCH. In *Proc. of HET-NETs Conference* (pp. 1-10).
- Shumway, R. H., Stoffer, D. S., Shumway, R. H., & Stoffer, D. S. (2017). ARIMA models. *Time series analysis and its applications: with R examples*, 75-163.
- Schreiber, I. (2009). Equities, Credits and Volatilities: A Multivariate AR-GARCH Analysis of the European Market.

Russel, E., Kesumah, F. S. D., Rialdi, A., & Usman, M. (2020). Dynamic modeling and forecasting stock price data by applying AR-GARCH model. *TEST Engineering and Management*, 82, 6829-6842.

Chen, X., & Hu, Y. (2022). Volatility forecasts of stock index futures in China and the US–A hybrid LSTM approach. *Plos one*, 17(7), e0271595.

Athey, S., & Imbens, G. W. (2019). Machine learning methods that economists should know about. *Annual Review of Economics*, 11, 685-725.