Journal of computer science and software applications
https://www.mfacademia.org/index.php/jcssa
ISSN:2377-0430
Vol. 4, No. 5, 2024

Hybrid LSTM-GARCH Framework for Financial Market Volatility Risk Prediction

Ke Xu[1], You Wu[2], Mohan Jiang[3], Wenying Sun[4], Zhirui Yang[5]

[1]Columbia University, New York, USA

[2]College of William & Mary, Williamsburg, USA

[3]New York University, New York, USA

[4]Southern Methodist university, Dallas, USA

[5]Columbia University, New York, USA

Correspondence should be addressed to You Wu; estellayouwu@gmail.com

Abstract:

This article explores a method that integrates deep learning with classical econometric models to address the challenge of predicting volatility risk in financial markets. In view of the limitations of traditional economic models in capturing complex financial market relationships, researchers propose a new framework that integrates long short-term memory networks (LSTM) and generalized autoregressive conditional heteroskedasticity models (GARCH). This framework takes full advantage of LSTM's ability to handle long-term dependencies and the GARCH model's advantages in capturing volatility and risk by introducing GARCH model parameters as input variables to the LSTM neural network. The experiment uses the historical data of the Nasdaq 100 Index for verification and compares the prediction effects of different models through a variety of evaluation indicators. The results show that the fusion model significantly outperforms the single LSTM model and other benchmark models in prediction accuracy.

Keywords:

Deep Learning; Long Short-term Memory Networks; GARCH; Financial Risk Management

1. Introduction

Financial risk management is crucial to maintaining market stability and promoting its healthy development, and volatility forecasting, as a key tool and core variable in risk management, plays an important role[1]. Given the complexity of the current financial system and its nonlinear structure, traditional econometric models have limitations in capturing these complex relationships[2]. Therefore, this study aims to explore an innovative method that combines deep learning algorithms with classic econometric models to solve the volatility prediction problem[3].

Specifically, we will build a comprehensive framework that integrates deep learning techniques and econometric principles[4]. Through this fusion, we are not only able to provide solid financial theoretical explanations but also effectively fit nonlinear characteristics in the data, thereby significantly improving forecast accuracy. The goal of this approach is to take full advantage of both methods to achieve more accurate and reliable volatility forecasts. This article first reviews the history and development status of financial market volatility forecast research. Next, the relevant theoretical basis and model construction methods are introduced in detail[5].

Subsequently, this paper adopts a hybrid approach to predict volatility. Specifically, we compare the performance of a single GARCH model and an LSTM model, and further propose seven GARCH-LSTM fusion models. The main innovations of this paper are:

1. Through the comparison of loss functions, this paper finds that when the GARCH model assumes that the residual follows the generalized error distribution (GED), its prediction accuracy is higher than that of other GARCH family models; the prediction accuracy of the LSTM model also exceeds that of the GARCH model;

2. In the fusion model, by adding GARCH model parameters to the LSTM model input, the fusion model based on quantity and price data shows better prediction performance than a single model; among them, the L-GT-GGED fusion model that combines three GARCH models and LSTM models performs best.

2. Related work

In the ever-changing financial market, the modeling and practical application of GARCH family models are constrained by strict assumptions. Therefore, how to organically integrate GARCH models with other emerging methods has become a hot topic. In volatility prediction, the research work using fusion models for prediction mainly includes: Kim et al. [6] (2018) took the volatility of KOSPI 200 as the research object and constructed a hybrid model GEW-LSTM of LSTM and three models of GARCH, EGARCH, and EWMA. Compared with other single models, this model has smaller mean square error (MSE) and mean absolute error (MAE), and has better stock market volatility prediction performance; Yan et al. [7] (2020) proposed a new hybrid method to predict copper price fluctuations, which not only combines deep neural networks with classic GARCH, but also combines LSTM with traditional artificial neural networks (ANN). Cao Wei et al. [8] (2020) constructed a hybrid model of LSTM and GARCH family to predict the volatility of RMB exchange rate. The experimental results show that the hybrid model performs better than the single model, and the single LSTM model performs better than the single GARCH model.

Verma [9] (2021) proposed a GARCH-GJR-LSTM hybrid model to predict crude oil volatility. Zeng et al. [10] (2022) combined deep learning methods with the generalized autoregressive conditional heteroskedastic mixed data sampling model (GARCH-MIDAS), used the GARCH-MIDAS model to deal with the heterofrequency problem between macroeconomic variables and stock market volatility, and predicted short-term volatility. Finally, the predicted short-term volatility was used as the input indicator of the deep learning model to predict the realized volatility of the stock market.The use of deep neural networks in financial market anomaly detection and risk assessment, as explored by Wang et al. [11], highlights the potential of deep learning in capturing complex patterns within financial data that traditional models may overlook. This work underscores the significance of deep neural networks in financial applications, particularly in enhancing the detection of anomalies that may signal underlying volatility risks.
Building on this foundation, Zheng et al. [12] introduced novel approaches to optimizing deep learning models by incorporating adaptive mechanisms such as sigmoid and tanh functions. These enhancements in optimization techniques can be particularly beneficial in the context of LSTM networks, where the long-term dependencies and non-linearities in financial time series data pose significant challenges. The improvements in optimizer performance, as detailed by Zheng et al., are crucial for refining the predictive capabilities of models like LSTM, which are central to the proposed hybrid LSTM-GARCH framework.

3.Method

Before introducing the fusion model, let’s first introduce the mathematical structure of the CARCH model.

$$y_t = x_t\gamma + \varepsilon_t$$

$$\sigma_t^2 = \omega + \sum_{i=1}^{q}\alpha_i\varepsilon_{i-1}^2 + \sum_{j=1}^{p}\beta_j\sigma_{t-j}^2$$

Where p represents the order of GARCH and q represents the order of ARCH. The condition for GARCH(p,q) to be stable is $\sum_{i=1}^{q} a_i + \sum_{j=1}^{p}\beta_j < 1$. And as $\sum_{i=1}^{q}\alpha_i + \sum_{j=1}^{p}\beta_j$ gradually approaches 1, the attenuation gradually slows down.

3.1. Introduction to LSTM

The Long Short Term Memory (LSTM) network is a relatively accurate network structure that integrates with the gradient learning algorithm. This is a variation of the RNN neural network that is currently widely used. It is almost the same as the ordinary RNN network structure in essence. LSTM just uses a different function to calculate the hidden layer state. The hidden layer of the traditional RNN model only has a state that is more sensitive to short-term information, while the hidden layer of the LSTM adds a state that has more storage capacity for long-term information. This state is similar to a "processor" that can determine whether to save or forget historical data, that is, the cell state. This improvement can well overcome the long-term dependence of RNN.

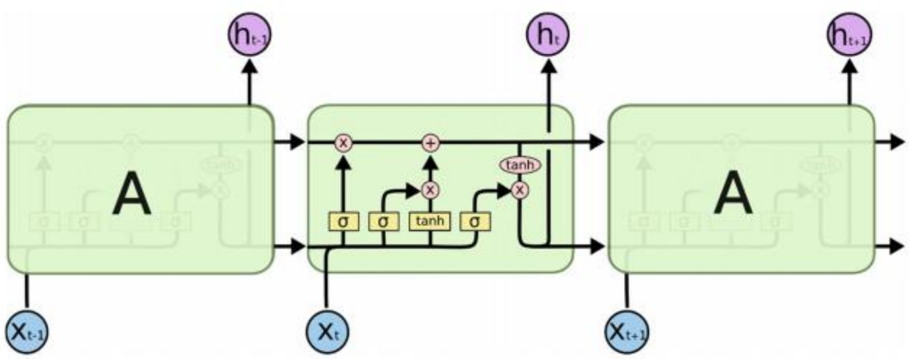

Figure 1. LSTM structure diagram

3.2. Fusion of GARCH and LSTM

This article intends to build a new fusion model by introducing the parameters of the GARCH family model and other explanatory variables as input variables of the LSTM neural network model to improve the prediction accuracy of volatility. In the GARCH model, the coefficient of the GARCH term represents the persistence of fluctuations, and the coefficient of the ARCH term represents the magnitude of the volatility shock. If the parameters of the GARCH model with different residual distributions are added to the LSTM neural network model, the hybrid model will be able to fully mine the volatility information contained in the parameters, thereby making the model prediction accuracy higher than that of a single LSTM model. significantly improved. Therefore, this paper first uses the GARCH family model for modeling, and uses its parameters as input variables of the LSTM model, so that the neural network model can effectively mine the volatility information in the time series and capture more volatility-related characteristics. Thereby improving the prediction performance of the model, its overall architecture is shown in Figure 2

Journal of computer science and software applications
https://www.mfacademia.org/index.php/jcssa
ISSN:2377-0430
Vol. 4, No. 5, 2024

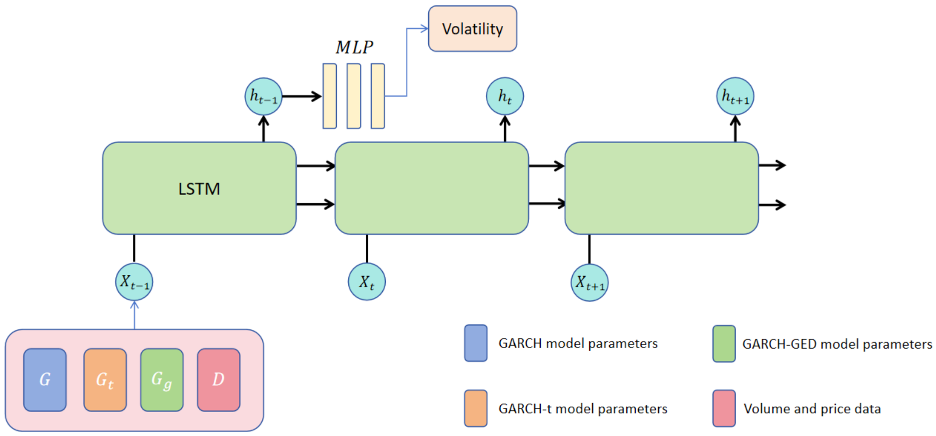

Figure 2. overall architecture

However, depending on the input GARCH model parameters, the constructed fusion model is also different. This paper establishes a total of 7 LSTM-GARCH models, and the specific model descriptions are shown in Table 1

Table 1: Parameters of 7 different models

Model Abbreviation	Notes
L-G	The LSTM model adds three parameters to the GARCH
L-GT	LSTM model adds three parameters of GT
L-GGED	LSTM model adds three parameters of GGED
L-G-GT	GARCH and GT parameter combination
L-G-GED	GARCH and GGED parameter combinations
L-GT-GGED	GT and GGED parameter combination
L-G-GT-GGED	All parameters

The specific steps of using LSTM-GARCH hybrid model to predict volatility are as follows:
The first step is to select and normalize the quantity and price data. The daily opening price, lowest price, highest price, and closing price of the research object are selected as the input variables of the quantity and price data, and the maximum and minimum standardization method is used to normalize the quantity and price data. The standardization method is shown in formula :

$$x_i^* = \frac{x_i - x_{\max}}{x_{\max} - x_{\min}}$$

x_i^* is the value of the variable x_i to be normalized after normalization, $x_{\max}$ represents the maximum value of the input variable, and $x_{\min}$ represents the minimum value of the input variable. Through normalization, the polarization of data can be avoided.
The second step is to obtain the parameters of the GARCH model. The estimation of the parameters of the GARCH family model is a sliding prediction process. With a window length of 15 days, the model parameters on day t are predicted based on the yield data from day t-16 to day t-1, and the model parameters on day t+1 are predicted based on the yield data from day t-15 to day t, and so on. The window slides backward to estimate the model parameters. The sliding prediction process is shown in Figure 3

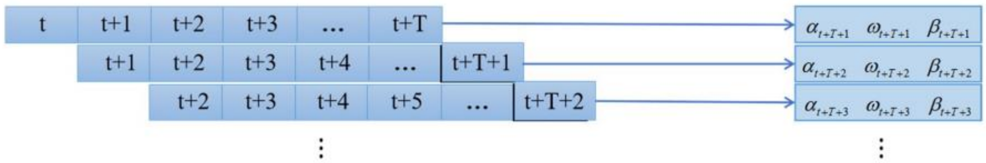

Figure 3. sliding prediction process

The third step is to build an LSTM neural network framework. Use pytorch to build the neural network framework, and use grid search and cross-validation methods to determine the optimal number of iterations, number of network layers and other parameter settings for the neural network.

The fourth step is to build a LSTM-GARCH hybrid model and perform model training. The GARCH model and the neural network related model are trained separately to obtain the volatility prediction results under different models.

4. Experiment

4.1. Evaluation indicators

Since a single error evaluation index has its limitations, if only one index is used to evaluate the model effect, it will produce a large deviation and it is difficult to make a comprehensive evaluation of the model's prediction results. Therefore, we use 6 evaluation indicators to evaluate the model, which are shown in the following table.

Table 2: Different evaluation indicators

Evaluation	Notes
MSE	$\frac{1}{T}\sum_{t=1}^{T}(RV_t-\delta_t^2)^2$
MAE	$\frac{1}{T}\sum_{t=1}^{T}\lvert RV_t-\delta_t^2\rvert^2$
RMSE	$\sqrt{\frac{1}{T}\sum_{t=1}^{T}(RV_t-\delta_t^2)^2}$
HMSE	$\frac{1}{T}\sum_{t=1}^{T}(1-\frac{\delta_t^2}{RV_t})^2$
HMAE	$\frac{1}{T}\sum_{t=1}^{T}\lvert 1-\frac{\delta_t^2}{RV_t}\rvert^2$
QLIKE	$\frac{1}{T}\sum_{t=1}^{T}(\ln\delta_t^2-\frac{RV_t}{\delta_t^2})^2$

4.2. Datasets

The dataset used in this article is the NASDAQ 100 dataset. This dataset is a widely used financial dataset that contains stock information of the 100 largest non-financial companies in the NASDAQ 100 index. These companies are usually leaders in the fields of science and technology and high technology, reflecting the market performance of US technology stocks. The dataset includes key indicators such as the opening price, highest price, lowest price, closing price and trading volume of stocks, covering many years of historical data. The dataset has been cleaned and the missing values have been filled, making it suitable for financial analysis tasks such as time series analysis and volatility prediction. Since the constituent stocks of the NASDAQ 100 index are high-growth and innovative, this dataset is of great value in studying market trends, developing forecasting models, and evaluating risk management strategies. The quantitative and price data features include four data: opening price, closing price, highest price, and lowest price. Then some results of GARCH model parameter estimation for some trading days are shown in Table 3.

Table 3: parameter estimation for some trading days

Data	ω_g	α_g	β_g	ω_t	α_t	β_t
2022.7.26	0.811	5.7e-16	0.746	2.599	0.184	8.2e-11
2022.7.27	1.553	9.5e-17	0.503	0.689	4.4e-15	0.779
2022.7.28	1.576	1.6e-16	0.503	2.582	0.165	1.1e-20
2022.7.29	1.968	0.352	0.878	1.974	0.352	1.3e-19
2022.7.30	1.529	0.454	0.551	1.534	0.453	0.932

After analyzing the data, it is divided into training set, validation set and test set in a ratio of 7:2:1. After the data set is divided, the LSTM model can be trained. Before training, the model parameters need to be set in advance. Considering that the parameter setting of the neural network model can greatly affect the training performance of the model, selecting a good set of parameters can improve the training performance and effect of the model. Therefore, this paper obtains the optimal parameter setting of the LSTM neural network model through the grid search and cross validation method as shown in Table 4. After obtaining the optimal parameters, the LSTM model and LSTM-GARCH fusion model can be trained using the optimal parameters.

Table 4: parameter estimation for some trading days

Hyperparameters	**epochs**	**Batch_size**	**optimizer**
LSTM	50	32	Adam

After training and predicting three GARCH models, LSTM models, and seven LSTM-GARCH models, the experimental results of the eleven models on the test set are shown in Table 5.

Table 5: parameter estimation for some trading days

Model	**MSE**	**MAE**	**RMSE**	**HMAE**	**HMSE**	**QLIKE**
GARCH	0.847	1.173	1.083	2.917	37.833	1.119
GT	0.846	1.158	1.076	2.861	36.424	1.111
GGED	0.780	1.259	1.029	2.513	30.063	1.079
LSTM	0.864	1.341	1.158	1.748	9.018	1.072
L-G	0.796	1.143	1.069	1.732	9.034	1.060
L-T	0.881	1.617	1.272	1.624	8.503	1.033
L-GG	0.751	1.234	1.111	1.476	9.847	1.055
L-G-T	0.846	1.423	1.193	1.561	8.805	1.043
L-G-GG	0.701	1.101	1.049	1.191	6.974	1.110
L-T-GG	0.686	1.081	1.040	1.452	7.687	1.086
Ours	0.604	1.107	1.056	1.073	5.133	1.071

Based on the provided experimental results, we can observe differences in the performance of different models on the prediction task. First of all, from the perspective of MSE, RMSE and other indicators, our model performs best among all compared models, with an MSE of 0.604, which is significantly lower than other models, which shows that our model has obvious advantages in prediction accuracy. . In addition,

from the perspective of HMAE and HMSE, although some models perform better on these indicators, our model still has the most comprehensive performance when considering all indicators comprehensively.
Further analysis, we can see that although traditional statistical models such as GARCH, GT and GGED have good performance on some indicators, their overall performance is not as good as deep learning models and combination models. In particular, although the LSTM model is slightly higher than our model in MSE, it performs worse in HMAE and HMSE, indicating that its prediction error is larger at some sample points.

5. Conclusion

This study presents a significant advancement in financial market volatility and risk prediction through the development of a hybrid LSTM-GARCH framework, effectively merging deep learning with classical econometric modeling. The fusion model leverages the GARCH model's ability to capture risk and the LSTM network's strength in processing sequential data and long-term dependencies, addressing the inherent complexities and nonlinearities in financial markets that traditional models struggle to encapsulate. Through rigorous empirical analysis using historical data from the Nasdaq 100 Index, the study demonstrates that the hybrid model consistently outperforms standalone models, including both the traditional GARCH variants and pure LSTM networks, across a range of critical performance metrics such as Mean Squared Error (MSE) and Root Mean Squared Error (RMSE). The superior predictive accuracy of the L-GT-GGED fusion model, which integrates multiple GARCH components with LSTM, underscores the value of combining quantitative and price data to enhance model performance. This research not only contributes to the field of financial risk management by providing a more accurate tool for volatility forecasting but also lays a solid foundation for future work in hybrid modeling, offering a novel approach that bridges the gap between econometric theory and machine learning in the context of financial markets.

References

[1] García-Medina A, Aguayo-Moreno E. LSTM–GARCH hybrid model for the prediction of volatility in cryptocurrency portfolios[J]. Computational Economics, 2024, 63(4): 1511-1542.

[2] Pan H, Tang Y, Wang G. A Stock Index Futures Price Prediction Approach Based on the MULTI-GARCH-LSTM Mixed Model[J]. Mathematics, 2024, 12(11): 1677.

[3] Liu S, Zhang Y, Wang J, et al. Fluctuations and Forecasting of Carbon Price Based on A Hybrid Ensemble Learning GARCH-LSTM-Based Approach: A Case of Five Carbon Trading Markets in China[J]. Sustainability, 2024, 16(4): 1588.

[4] Mualifah L N A, Soleh A M, Notodiputro K A. Comparison of GARCH, LSTM, and Hybrid GARCH-LSTM Models for Analyzing Data Volatility[J]. Int. J. Advance Soft Compu. Appl, 2024, 16(2).

[5] Ray S, Lama A, Mishra P, et al. An ARIMA-LSTM model for predicting volatile agricultural price series with random forest technique[J]. Applied Soft Computing, 2023, 149: 110939.

[6] G.Kim H, Won C. Forecasting the Volatility of Stock Price Index: A Hybrid Model Integrating LSTM with Multiple GARCH-Type Models[J]. Expert Systems with Applications, 2018, 103(aug.):25-37

[7] Yan H, Jian N, Liu W. A hybrid deep learning approach by integrating LSTM-ANN networks with GARCH model for copper price volatility prediction[J]. Physica A: Statistical Mechanics and its Applications, 2020,557

[8] Cao Wei, Ren Siru. Research on RMB exchange rate fluctuation prediction based on LSTM and GARCH family hybrid model[J]. Journal of Computer Application Research, 2020, 37(S1):79-82.

[9] Verma S. Forecasting volatility of crude oil futures using a GARCH–RNN hybrid approach[J]. Intelligent Systems in Accounting, Finance and Management, 2021,28(2).

[10] Zeng H, Shao B, Bian G, Dai H, Zhou F. A hybrid deep learning approach by integrating extreme gradient boosting-long short-term memory with generalized autoregressive conditional heteroscedasticity family modelsfor natural gasload volatility prediction[J]. Energy Science & Engineering,2022,10(7)

[11] Wang, B., Dong, Y., Yao, J., Qin, H., & Wang, J. (2024). Exploring Anomaly Detection and Risk Assessment in Financial Markets Using Deep Neural Networks. International Journal of Innovative Research in Computer Science & Technology, 12(4), 92-98.

[12] Zheng, H., Wang, B., Xiao, M., Qin, H., Wu, Z., & Tan, L. (2024). Adaptive Friction in Deep Learning: Enhancing Optimizers with Sigmoid and Tanh Function. arXiv preprint arXiv:2408.11839.