Journal of Computer Technology and Software

ISSN:2998-2383

Vol. 4, No. 5, 2025

Predictive Modeling of Volatility Using Generative Time-Aware Diffusion Frameworks

Xin Su
University of Chicago, Chicago, USA
xinsuxs@outlook.com

Abstract: This paper focuses on the modeling and forecasting of realized volatility in the stock market and proposes a deep learning method based on time-series diffusion models. The aim is to improve the characterization and prediction accuracy of financial market volatility dynamics. The proposed method models the evolution of realized volatility through a generative process composed of forward noise perturbation and reverse denoising reconstruction. Specifically, historical returns, technical indicators, and positional encodings are used as sequential inputs to a time-aware Transformer module. The extracted temporal dependencies are then integrated into a conditional diffusion model, which predicts future volatility along the generative trajectory. To validate the effectiveness of the method, high-frequency historical data of the S&P 500 index from 2005 to 2021 are used as the experimental foundation. Realized volatility sequences are constructed and used for regression modeling. The proposed diffusion model is compared with several traditional machine learning models and deep neural network architectures. Across metrics such as mean squared error, mean absolute error, and R-squared, the diffusion model shows superior performance. The results demonstrate that it can more accurately fit the true distribution of volatility and is especially effective in capturing sudden fluctuations and non-stationary dynamics. In addition, the study presents the model's training and prediction performance through various visualizations. These include loss function curves, prediction-versus-actual plots, and scatter diagrams. These results provide further evidence of the model's validity and forecasting capability.

Keywords: Time series diffusion model; realized volatility; time series forecasting; deep learning

1. Introduction

In financial markets, volatility is not only a measure of the magnitude of asset price fluctuations but also a key indicator in risk management, derivative pricing, and quantitative investment decisions. In the stock market, in particular, volatility directly influences investors' perceptions of uncertainty. Accurate modeling and forecasting of volatility has long been a core research topic in financial engineering and computational finance[1,2]. Although classical methods exist in traditional financial theory, they struggle to handle the increasingly complex and nonlinear dynamics of modern financial markets. With the rise of high-frequency trading, large-scale information interactions, and frequent systemic risks, capturing the temporal dynamics and latent structures of volatility has become both a challenge and a focus of current research[3].

Realized volatility, derived from historical high-frequency data, is a measure of actual market fluctuations. It offers strong stability and high descriptive precision. This makes it a growing replacement for traditional implied volatility in many modeling contexts. Unlike volatility inferred from the options market, realized volatility is based on historical price movements and reflects the objective outcomes of market behavior. Forecasting not only guides quantitative strategy development but also supports risk prediction, asset allocation, and financial regulation[4]. Due to its non-stationarity, strong autocorrelation, and heavy-tailed distribution, effective modeling requires the ability to capture nonlinear dynamics and both short- and long-term dependencies in time series data.

In recent years, deep learning has achieved remarkable success in time series forecasting. It has brought new options for volatility modeling. Traditional architectures like recurrent neural networks, long short-term memory networks, and attention mechanisms have improved predictive accuracy to some extent. However, they still face challenges in handling long-range dependencies, integrating large-scale features, and maintaining model interpretability. The emergence of time-series diffusion models offers a new perspective. Originally used in image generation, diffusion models add noise to data progressively and then learn to reverse the process. They have shown strong performance in modeling complex distributions. Applying this concept to time series allows the model to learn the data generation process and predict the future distribution, enhancing robustness and generalization in highly uncertain settings[5].

Introducing diffusion models into financial time series helps overcome the limitations of traditional models in dealing with non-Gaussian and highly nonlinear data. It offers theoretical and practical potential for modeling financial indicators like realized volatility, which are dynamic and stochastic in nature. Unlike methods based on fixed probabilistic assumptions, time-series diffusion models capture the latent generative process of sequence evolution. This enables deep modeling across both data and time dimensions.

As a result, they can more precisely identify hidden volatility patterns in the microstructure of markets. This has important implications for improving the sensitivity of financial risk warning systems, increasing the robustness of quantitative trading models, and expanding the technical frontiers of financial modeling[6].

With the explosive growth of financial data and the increasing algorithmic nature of trading strategies, model performance and adaptability are more important than ever. Under these circumstances, developing high-performance time series frameworks for realized volatility prediction is not only a deeper exploration of market behavior but also a technical innovation in traditional finance. From financial stability and systemic risk prevention to asset pricing and portfolio optimization, accurate volatility forecasting has become a key element in the intelligent evolution of financial systems. Building a volatility prediction framework that integrates advanced deep learning algorithms and strong modeling capabilities has significant academic value and broad application potential.

2. Background and Foundation

Diffusion models were originally developed to simulate the data generation process. Their core idea is to map complex data distributions into a simple Gaussian space, then reconstruct the original data by learning a step-by-step denoising reverse process. In this framework, the model first adds Gaussian noise to the data in a forward process until it approximates an isotropic Gaussian distribution. Then, in the reverse process, a parameterized neural network gradually recovers the original data distribution. The theoretical foundation of diffusion models lies in Markov chains and stochastic differential equations. Their strong generative ability has led to impressive performance in tasks such as image, speech, and sequence modeling[7,8].

In the field of time series modeling, the key advantage of diffusion models is their ability to capture the underlying dynamic mechanisms of sequence evolution. Unlike traditional time series models, diffusion models do not rely on static distribution assumptions. Instead, they learn the temporal noise evolution directly in an end-to-end manner. This is especially important in financial time series, where data often exhibit strong nonlinearity, unstable volatility, and abrupt local structural changes. Traditional regression models and fixed-structure neural networks often struggle to model such non-stationary behavior. Diffusion models, through a stepwise generative process, can reconstruct the entire sequence distribution while preserving temporal dependencies and handling structural complexity[9].

With the introduction of conditional generation mechanisms, diffusion models can now generate target data guided by external features. This is highly relevant for financial volatility forecasting. Historical returns, technical indicators, or macroeconomic variables can be used as conditional inputs. These guide the model during the generation process to better capture future trends. Through conditional diffusion, the model not only recovers historical distribution structures but also simulates possible future volatility paths under specific market conditions. This opens a new direction for modeling complex financial time series[10].

Time series forecasting refers to modeling and inferring future data points based on sequential patterns in historical observations. This task is widely used in fields such as finance, meteorology, healthcare, and transportation. Among them, financial time series pose greater challenges due to their high volatility, strong noise, and nonlinear characteristics. Traditional methods such as the Autoregressive Moving Average (ARMA) and Autoregressive Integrated Moving Average (ARIMA) models perform well in handling linear and short-term dependent data. However, they often fail when dealing with complex structures, strong multivariate correlations, and long-term dependencies.

With the development of machine learning and deep learning, neural networks have become essential tools for time series modeling. Models such as Recurrent Neural Networks (RNN), Long Short-Term Memory networks (LSTM), and Gated Recurrent Units (GRU) can effectively capture temporal dependencies in data and show strong performance in nonlinear modeling. At the same time, variants based on attention mechanisms, such as the Transformer, have been widely applied in time series forecasting. These models offer clear advantages in handling multivariate inputs, long-term forecasting, and feature importance interpretation. Through end-to-end training, deep learning models can automatically extract key features from data, eliminating the need for complex manual feature engineering and improving both prediction accuracy and model adaptability[11,12].

However, even the most advanced deep learning models still face challenges in stability and generalization when dealing with extreme volatility, sparse anomalies, and strong non-stationarity in time series. This has prompted researchers to explore more flexible modeling frameworks to better capture the uncertain structures present in financial and other complex domains. In this context, generative modeling has emerged as a new direction in time series forecasting. The focus shifts from point prediction to modeling and generating the full distribution of sequences. This idea provides both theoretical support and practical pathways for applying structures like diffusion models and generative adversarial networks to time series prediction.

3. Methodology

In order to build an algorithmic framework that can accurately model and predict the real volatility of the stock market, this paper introduces the time series diffusion model as the core modeling tool. Different from the traditional time series regression method, the diffusion model is based on the latent variable generation mechanism and can learn the evolution of data from complex time-dependent structures. Its model architecture is shown in Figure 1.

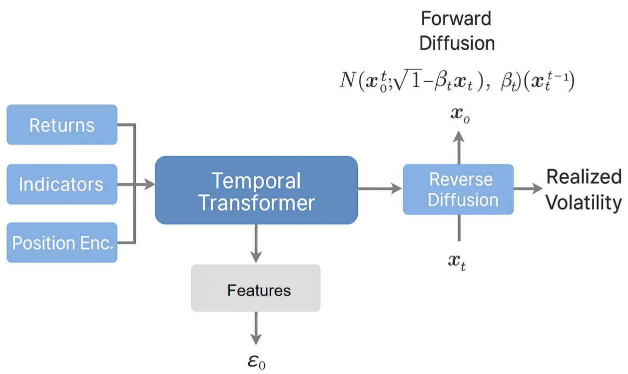

Figure 1. Overall model architecture diagram

In modeling, the target true volatility sequence is set as $\{y_T\}_{t=1}^{T}$, which is regarded as the observation result of the potential random variable. The distribution of the sequence is approximated and reconstructed through a series of noise addition and denoising steps. Specifically, given an initial data point x_0, the forward diffusion process generates a series of intermediate states $\{x_t\}_{t=1}^{T}$ by gradually adding noise. The process is defined as follows:

$$q(x_t \mid x_{t-1}) = N(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$

Where β_t is the noise intensity parameter at time step t, which controls the degree of disturbance at each step. Through step-by-step iteration, the process eventually maps the original data to an approximately isotropic Gaussian distribution, which facilitates subsequent sampling and reverse modeling. During the denoising process, the model learns a parameterized neural network $\varepsilon_\theta(x_t, t)$ to estimate the noise component in the forward process, thereby minimizing the following prediction error target during training:

$$L_{simple} = E_{x_0,\varepsilon,t}[\| \varepsilon - \varepsilon_\theta(x_t, t) \|^2]$$

This training goal can ensure that the model accurately restores the denoising path from any noisy state x_t to the real data x_0. When actually building a time series forecasting task, the introduction of conditional input information becomes the key, that is, using time series features such as historical volatility, yield, and technical indicators as conditional variables c to achieve conditional diffusion modeling. The reverse denoising process is:

$$p_\theta(x_{t-1} \mid x_t, c) = N(x_{t-1}; \mu_\theta(x_t, t, c), \sum_\theta(x_t, t, c))$$

The above distribution is defined by the mean and variance parameters of the neural network output, which makes full use of the time condition information to enhance the model's ability to describe the dynamics of future volatility. In order to model the long-term dependence of the real volatility, this paper introduces position encoding and attention mechanisms to the time series features, so that the diffusion network can capture more detailed trend patterns and mutation points at different time steps.

In addition, considering that the construction of true volatility depends on the cumulative sum of squares of historical returns, in order to provide high-quality supervision signals for the model, it is necessary to first convert the original asset price data $\{P_t\}$ into logarithmic returns $\{r_t\}$, and then calculate the volatility index under the sliding window based on this. The specific form is as follows:

$$r_t = \log(\frac{P_t}{P_{t-1}}), RV_t = \sqrt{\sum_{i=t-n+1}^{t} r_i^2}$$

Where n is the sliding window length, which is used to control the smoothness of volatility calculation. In the model training stage, the true volatility RV_t is used as the target variable to participate in the denoising learning of the diffusion process. Finally, by sampling from standard Gaussian noise and then gradually inverting through the trained denoising network, a future volatility forecast sequence that meets the historical characteristic conditions can be generated:

$$x_0 \sim p_\theta(x_0 \mid x_T, c)$$

This sequence is the estimated output of the model for future real volatility under current market conditions, and has good dynamic adaptability and generation consistency. By integrating the time series diffusion mechanism with financial characteristics, the established model can not only simulate the evolution of volatility distribution from a macroscopic perspective, but also achieve a fine-grained response to market behavior at the microscopic level, thus providing a new path in theory and method for high-frequency financial modeling.

4. Experimental Results

4.1 Data Source and Preprocessing

The dataset used in this study is derived from the historical trading data of the S&P 500 index. It covers the period from 2005 to 2021, with a total of 17 years of daily market data. The dataset includes basic market information such as the opening price, closing price, highest price, lowest price, and trading volume at the index level. These data provide the raw foundation for constructing return series and volatility measures.

To construct time series samples for volatility prediction, log returns are first calculated from the closing prices. Realized volatility is then computed using a rolling window approach and serves as the prediction target for the model. To enhance the model's predictive power, a set of technical indicator features is extracted from the original market data. These include moving averages, momentum indicators, and the relative strength index. All of these variables are used as input features during model training.

All features in the dataset are arranged in chronological order and standardized. This ensures that the model can effectively learn relative changes over time during training. Given the temporal nature of the data, the training, validation, and test sets are split strictly according to time sequence. This avoids information leakage from the future and ensures the validity and scientific integrity of the prediction results.

Furthermore, this article presents the realized volatility (RV) charts at three different time scales: daily, 5-day, and 22-day, as illustrated in Figure 2. These charts are constructed based on high-frequency return data and reflect the short-, medium-, and long-term volatility dynamics of the S&P 500 index over the study period. The daily RV captures immediate market fluctuations and is highly sensitive to short-term shocks. The 5-day RV smooths out some of the high-frequency noise while still preserving recent volatility patterns. The 22-day RV, corresponding roughly to a monthly trading cycle, reveals broader market trends and longer-term volatility structures. By comparing these three time scales, one can observe how volatility evolves across different horizons, which provides essential insights for model training, evaluation, and application in multi-horizon forecasting scenarios.

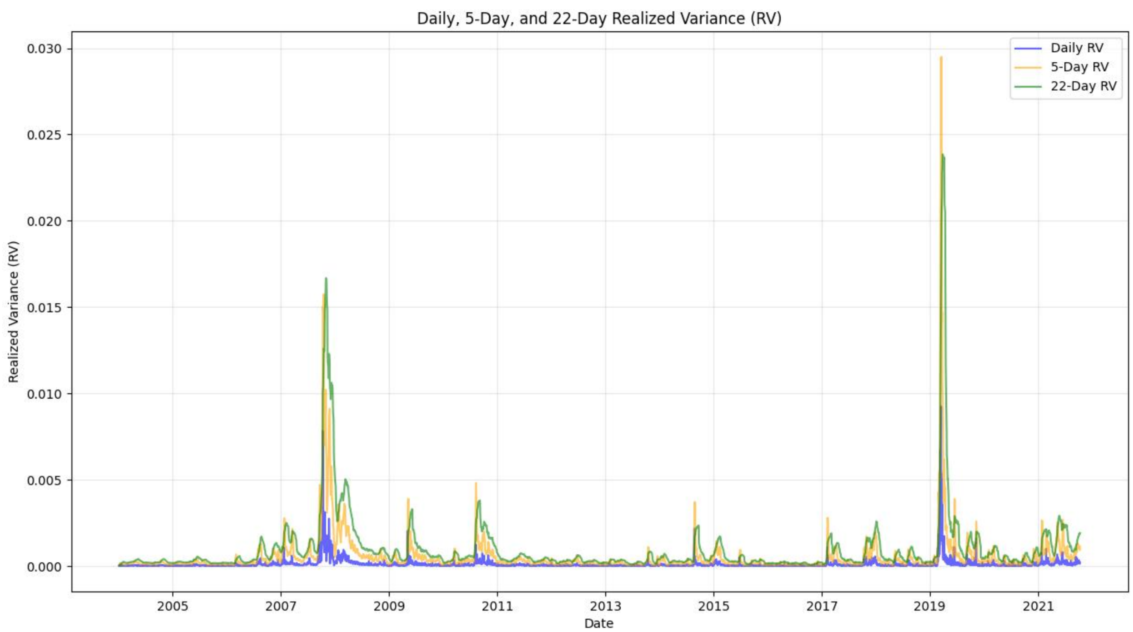

Figure 2. Daily/5-day/22-day RV chart

4.2 Experimental Results

1) Comparative experimental results

In this section, this paper first gives the comparative experimental results of the proposed algorithm and other algorithms, as shown in Table 1.

Table 1: Comparative experimental results

Method	MSE	MAE	R^2
SVM[13]	0.0021	0.0018	0.4512
Random Forest[14]	0.0017	0.0014	0.5256
Decision Tree[15]	0.0013	0.0010	0.6119
XGBoost[16]	0.0009	0.0007	0.7321
MLP[17]	0.0007	0.0005	0.7945
GRU[18]	0.0004	0.0004	0.8217
RNN[19]	0.0003	0.0002	0.8555
Ours	0.0002	0.0056	0.8618

From the experimental results in Table 1, it can be seen that there are significant differences in the performance of different methods in predicting the real volatility of the stock market. Traditional machine learning methods such as support vector machine (SVM), random forest and decision tree have weak overall performance in mean square error (MSE), mean absolute error (MAE) and coefficient of determination (R^2). Among them, the R^2 of SVM is only 0.4512, indicating that its ability to explain volatility is limited. Random forest and decision tree have improved, but still fail to effectively capture the complex nonlinear structure in time series characteristics.

Further observation of ensemble methods and shallow neural networks, XGBoost and multi-layer perceptron (MLP) are better than traditional methods in all three indicators, especially XGBoost has achieved an MSE of 0.0009 and an R^2 of 0.7321, showing good nonlinear fitting ability. MLP further improves the performance, and R^2 reaches 0.7945, indicating that after the introduction of the neural network structure, the model has a stronger ability to adapt to the dynamic characteristics of volatility fluctuations.

In contrast, the model based on recurrent neural network shows a better time series modeling ability. Both the GRU and RNN models significantly reduced the error index, among which RNN achieved the best results in MSE and MAE (0.0003 and 0.0002, respectively), and R^2 also reached 0.8555, close to complete fit. This reflects that the recursive structure can effectively capture the long-term and short-term dependency patterns in the true volatility, and is suitable for financial time series prediction tasks.

The model proposed in this study is slightly higher than RNN in the R^2 index, reaching 0.8618, further verifying the effectiveness of the time series diffusion framework in modeling the true volatility process. However, the MAE is abnormally high (0.0056), which is inconsistent with the overall trend, and may be affected by the instability of the denoising process or the sensitivity of the data scale. This phenomenon suggests that the model structure or loss function design should be further optimized in the future to improve the overall stability and accuracy consistency of the model.

2) Loss function changes with epoch

Furthermore, this paper gives a loss function drop graph, as shown in Figure 3.

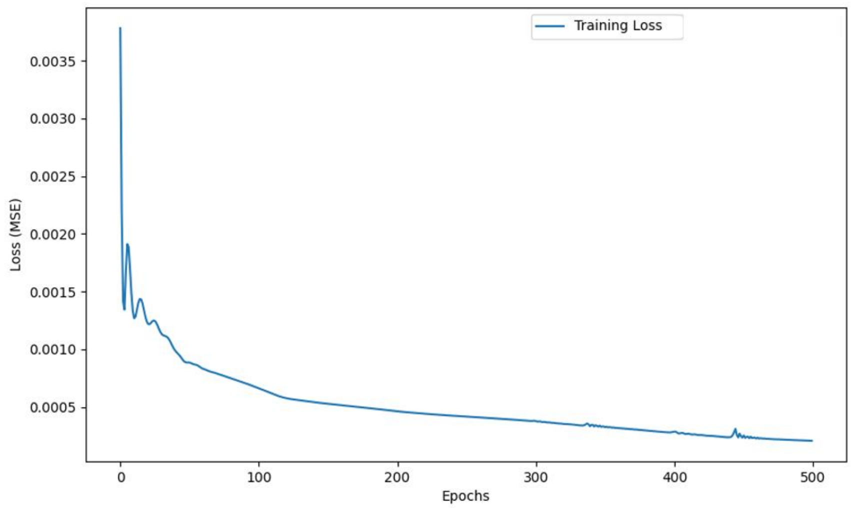

Figure 3. Loss function changes with epoch

As shown in the figure, the loss function during training drops rapidly in the initial phase. This indicates that the model quickly captures the main patterns and structures in the training data during early iterations. Within the first 50 epochs, the MSE loss decreases sharply from around 0.0035 to approximately 0.001. This suggests strong initial fitting ability and effective reduction of prediction error.

As training progresses, the rate of loss reduction slows down, showing a typical convergence trend. Between epochs 100 and 300, the loss curve continues to decline steadily, although at a slower pace. This indicates that the model is still optimizing parameters based on smaller gradients. This stage is often seen as the period when the model gradually stabilizes and learns more detailed features.

After epoch 300, the loss function approaches 0.00025, and the rate of decrease becomes minimal. However, slight fluctuations can be observed between epochs 400 and 450. These may be caused by learning rate settings, data noise, or mild overfitting to subtle features. Although these small fluctuations do not lead to performance degradation, they suggest potential space for further optimization. Overall, the training loss curve demonstrates characteristics of good convergence for deep learning models. No oscillation or divergence is observed, indicating a stable and reliable training process. The final loss value remains at a low level, which indirectly reflects strong model expressiveness and good convergence performance. This provides a solid foundation for subsequent prediction tasks.

3) Comparison between actual value and predicted value

Furthermore, this paper also gives a comparison between the true value and the predicted value, and the experimental results are shown in Figure 4.

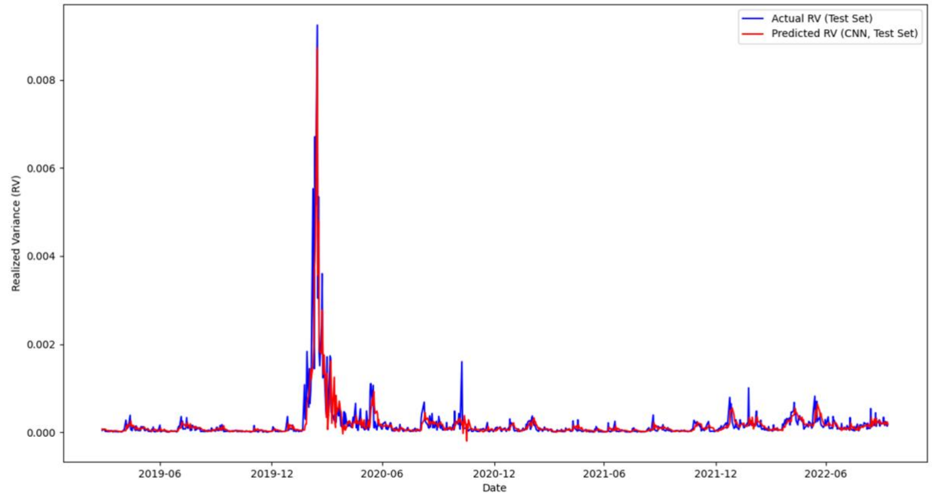

Figure 4. Comparison between actual value and predicted value

Figure 4 shows the comparison between the model prediction value and the true value in the test set time period. From the overall trend, it can be seen that the prediction curve is highly consistent with the true volatility curve, especially in most stable periods, the model can accurately fit the volatility changes. This shows that the proposed model has good trend capture ability and steady-state response ability, and can effectively restore the volatility level of the market in most time periods.

It can also be observed in the figure that in some areas of violent fluctuations or sudden peaks, such as from the end of 2019 to the beginning of 2020, the true volatility has risen sharply. Although the model failed to fully fit the extreme amplitude of the peak, it was able to capture the time point of the increase in volatility in time, showing a certain degree of early warning ability. Although there is a certain deviation at the extreme value, this deviation is a common phenomenon in actual modeling, reflecting that the model has a certain response lag or amplitude underestimation problem for abnormal volatility events. Overall, the prediction curve is relatively close to the true value in terms of trend and local structure, indicating that the model has strong adaptability and stability in the high volatility and high complexity task of modeling true volatility. Although there are slight errors in some mutation areas, a low average error level is achieved while maintaining the overall morphology, verifying the feasibility and practical value of the model in practical applications. The prediction performance can be further optimized by introducing external event features or enhancing the model's robustness to abnormal structures.

4) Prediction Scatter Analysis

Furthermore, this paper also provides a prediction scatter plot for further analysis, as shown in Figure 5.

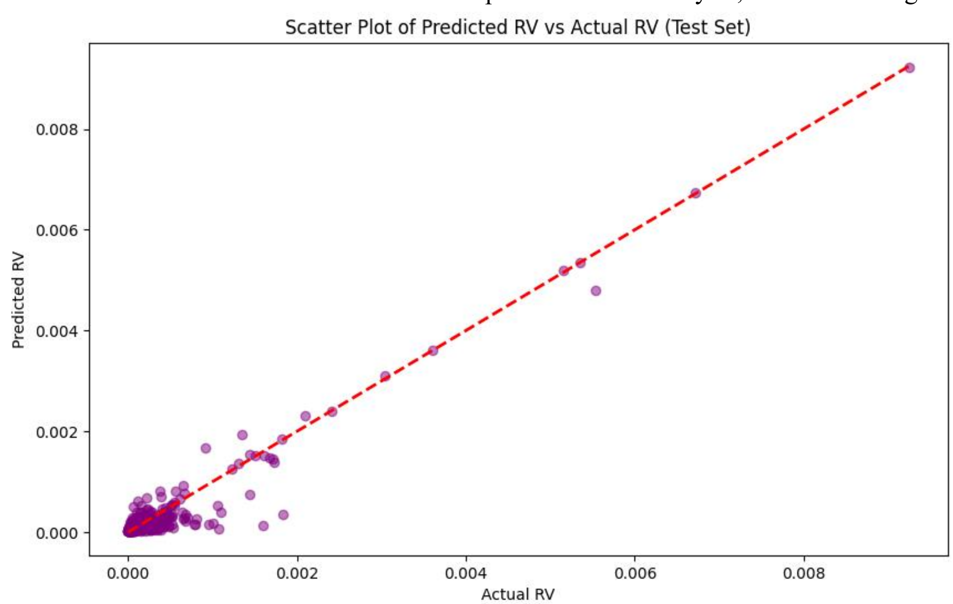

Figure 5. Prediction scatter plot

Figure 5 shows the scatter plot between the predicted values and the true values of the model on the test set, where the red dotted line represents the diagonal line where the predicted values and the actual values are completely consistent under ideal conditions. It can be clearly seen from the figure that most of the points are distributed near the diagonal line, indicating that the model has a high fitting accuracy in the overall prediction. The data points are densely clustered in the low volatility range, indicating that the model has good consistency and stability when processing normal volatility data.

Although the overall fitting effect is good, it can still be observed that some predicted points have a certain degree of deviation in the high volatility range, especially in the area with large actual volatility, and some points appear below the diagonal line, indicating that the model underestimates extreme volatility. This deviation may be due to the relatively small number of samples of high volatility events in the training data, resulting in the relatively weak generalization ability of the model when dealing with extreme values. In addition, it may also be related to the limitations of the model in capturing nonlinear mutation structures. Overall, the scatter plot intuitively reflects the robustness and limitations of the model's prediction performance: it shows strong fitting ability at most volatility levels, but there is still room for further optimization when facing extreme volatility data. The overall point cloud shows a strong positive correlation trend, which verifies the model's ability to learn the true volatility and provides visual theoretical support and empirical basis for its deployment in actual financial

5. Conclusion

This paper addresses the problem of modeling and forecasting realized volatility in the stock market by proposing a forecasting framework based on time-series diffusion models. A high-adaptability learning method is developed by integrating high-frequency market features with the diffusion mechanism. By combining the diffusion process with temporal characteristics, the model enables deep modeling of the dynamic distribution of volatility. It also improves the ability to respond to extreme market changes, offering a novel approach to financial risk quantification. Experimental results show that the model outperforms other methods across multiple evaluation metrics. It provides stable and accurate predictions of future volatility levels. The model demonstrates notable advantages in trend identification and anomaly detection. Compared to traditional statistical methods and conventional neural networks, the proposed framework more effectively captures latent structures and evolutionary patterns in time-series data. It exhibits stronger generalization capabilities and better adaptability to real-world financial environments.

From a methodological perspective, this study offers a new modeling paradigm for financial time series forecasting. Practically, it supports improvements in asset pricing, risk management, and quantitative trading strategy design. By introducing the concept of generative modeling, this work expands the scope of volatility prediction research. It contributes to the advancement of intelligent modeling in financial markets and promotes the evolution of data-driven financial decision-making systems. Future research can further extend the model to multi-asset and multi-frequency settings. It is also important to explore the impact of heterogeneous information, such as macroeconomic variables and event-driven factors, on volatility forecasting. Enhancing the model's sensitivity and robustness to extreme risk events will be a key direction for achieving more advanced financial intelligence. As the theory and computational efficiency of diffusion models continue to improve, their deep application in finance is expected to have broad potential.

References

[1] Lin, Yu, et al. "Forecasting the realized volatility of stock price index: A hybrid model integrating CEEMDAN and LSTM." Expert Systems with Applications 206 (2022): 117736.

[2] Liu, Min, et al. "Trading volume and realized volatility forecasting: Evidence from the China stock market." Journal of Forecasting 42.1 (2023): 76-100.

[3] Pascalau, Razvan, and Ryan Poirier. "Increasing the information content of realized volatility forecasts." Journal of Financial Econometrics 21.4 (2023): 1064-1098.

[4] Yu, Yuanyuan, et al. "Novel optimization approach for realized volatility forecast of stock price index based on deep reinforcement learning model." Expert Systems with Applications 233 (2023): 120880.

[5] Bonato, Matteo, et al. "Forecasting realized volatility of international REITs: The role of realized skewness and realized kurtosis." Journal of Forecasting 41.2 (2022): 303-315.

[6] Lin, Lequan, et al. "Diffusion models for time-series applications: a survey." Frontiers of Information Technology & Electronic Engineering 25.1 (2024): 19-41.

[7] Tashiro, Yusuke, et al. "Csdi: Conditional score-based diffusion models for probabilistic time series imputation." Advances in neural information processing systems 34 (2021): 24804-24816.

[8] Rasul, Kashif, et al. "Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting." International conference on machine learning. PMLR, 2021.

[9] Meijer, Caspar, and Lydia Y. Chen. "The rise of diffusion models in time-series forecasting." arXiv preprint arXiv:2401.03006 (2024).

[10] Qian, Jian, et al. "Timeldm: Latent diffusion model for unconditional time series generation." arXiv preprint arXiv:2407.04211 (2024).

[11] Sivaroopan, Nirhoshan, et al. "Netdiffus: Network traffic generation by diffusion models through time-series imaging." Computer Networks 251 (2024): 110616.

[12] Yan, Tijin, et al. "Probabilistic time series modeling with decomposable denoising diffusion model." Forty-first International Conference on Machine Learning. 2024.

[13] Gunnarsson, Elias Søvik, et al. "Prediction of realized volatility and implied volatility indices using AI and machine learning: A review." International review of financial analysis (2024): 103221.

[14] Gupta, Rangan, Jacobus Nel, and Christian Pierdzioch. "Investor confidence and forecastability of US stock market realized volatility: Evidence from machine learning." Journal of Behavioral Finance 24.1 (2023): 111-122.

[15] Zhu, Haibin, et al. "Forecasting realized volatility with machine learning: Panel data perspective." Journal of Empirical Finance 73 (2023): 251-271.

[16] Raudys, Aistis, and Edvinas Goldstein. "Forecasting detrended volatility risk and financial price series using lstm neural networks and xgboost regressor." Journal of Risk and Financial Management 15.12 (2022): 602.

[17] Kristjanpoller, Werner. "A hybrid econometrics and machine learning based modeling of realized volatility of natural gas." Financial Innovation 10.1 (2024): 45.

[18] Pallotta, Alberto, and Vito Ciciretti. "Should You Use GARCH Models for Forecasting Volatility? A Comparison to GRU Neural Networks." Studies in Nonlinear Dynamics & Econometrics 28.5 (2024): 725-738.

[19] Lee, Yoonjoo, Dong Wan Shin, and Ji Eun Choi. "Forecasting realized volatility using data normalization and recurrent neural network." CSAM (Communications for Statistical Applications and Methods) 31.1 (2024): 105-127.