Expert Systems With Applications 238 (2024) 121968

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Improving prediction efficiency of Chinese stock index futures intraday price by VIX-Lasso-GRU Model

Wen Fang [*], Shuwen Zhang, Chang Xu

Department of Finance, School of Economics and Management, Beijing Jiaotong University, Beijing 100044, PR China

ARTICLE INFO

Keywords:
Stock index future price
Prediction
VIX
GRU
Lasso algorithm

ABSTRACT

With $T + 0$ and short selling mechanism, the stock index futures are attractive to short-term traders in China, where stocks cannot be liquidated within the day and are difficult to short. So in terms of futures, how to improve the accuracy and speed of intraday price forecasting always fascinates short-term traders and researchers. Here we propose a novel forecasting model, VIX-Lasso-GRU Model, which based on the gated recurrent unit (GRU) by adding VIX information and a method called Least absolute shrinkage and selection operator (Lasso). The volatility index (VIX) can reduce the prediction errors and the Lasso algorithm significantly improve the training speed of the model. We predict the 5-minute closing prices of three datasets of index futures by VIX-Lasso-GRU Model. Comparing to the pure GRU and LSTM, we find that this new prediction model can improve the prediction efficiency with faster speed and higher accuracy.

1. Introduction

Stock index futures are derived from the high development of the capital market, providing an effective way for investors to hedge the systematic risk of stocks. The basic arrangement for hedging risk in futures market is the short-selling mechanism, which is especially important in the Chinese market because of the difficulties in shorting a certain stock. Launched in 2010, the development of the stock index futures market in China is rapid, and its turnover accounted for 15.6% of the total futures transaction in 2021. But the existing research shows that the market deviates from the effective market hypothesis, which verifies the possibility of predicting price (Chen & Pan, 2016; Li, Shen, Wang, & Zhang, 2020). Based on such results, this paper proposes a new model for futures price prediction and we verify its effectiveness on three datasets of stock index futures, including Shanghai Stock Exchange 50 (SSE 50) index futures, Shanghai Shenzhen 300 (CSI 300) index futures and Shanghai Shenzhen 500 (CSI 500) index futures. A good prediction model is of great significance for investors to increase returns and avoid risks. It can also be used to develop investment strategies.

Regarding of price forecasting, a large number of models have been established, mainly divided into parametric models and non-parametric ones, which is led by artificial intelligence (AI). Including ARMA, ARIMA, GARCH and their variants or hybrids, parametric models are built based on assumptions of certain probability distribution. They have been widely applied in predictions of financial time series (Bhardwaj & Swanson, 2006; Vo & Ślepaczuk, 2022; Yu & Huang, 2021). However, there are disadvantages of parametric models, such as strict distribution assumptions and difficulties in dealing with massive data, which are exactly incompatible with the characteristics of financial data. On the contrary, AI models are good at handling time series with large amount of data and uncertain distribution. Ding, Cui, Xiong, and Bai (2020) built return prediction models based on genetic programming, proving that these models have better prediction performance than AR family models. Nowadays, neural network is the mainstream branch of AI models in prediction. At first, simple artificial neural networks (ANN) are used for predictions, such as back propagation neural network (BP), radial basis function (RBF) and multi-layer perceptron neural network (MLP) (Li et al., 2016; Qiu, Song, & Akagi, 2016; Zhang & Wu, 2009). With deeper understanding of capital market, it is found that financial time series are usually long-range correlated, which means that the past information has an impact on the current state. Thus recurrent neural networks (RNN), especially ones with gated control structures such as the long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) and GRU (Cho et al., 2014), which can automatically filter memories, have become the mainstream of forecasting financial data. The empirical results from different nations and markets show that the prediction accuracy of RNN is higher than that of both ANN and parametric models (Cao, Li, & Li, 2019; Fischer & Krauss, 2018; Jiang, Liu, Zhang, & Liu, 2020; Kim & Won, 2018; Wu, Wang, Su, Tang, & Wu, 2020; Zhao, Zeng, Liang, Kang, & Liu, 2021).

* Corresponding author.
E-mail addresses: wenfang@bjtu.edu.cn (W. Fang), shuwenzhang@bjtu.edu.cn (S. Zhang), 18241101@bjtu.edu.cn (C. Xu).

https://doi.org/10.1016/j.eswa.2023.121968
Received 26 April 2023; Received in revised form 12 September 2023; Accepted 1 October 2023
Available online 5 October 2023

In this paper, we choose GRU as the basic model, which can better meet the requirements of fast processing than LSTM (We also conducted experiments based on LSTM and demonstrated the rationality of choosing GRU and the details of experiments are presented in Section 4.2), especially when it comes to high frequency data. And then we creatively bring up two ways to optimize the model, one is to increase the effective input information while the other is to reduce the data redundancy. Asset prices can be decomposed into trends and fluctuations, both of which should be considered while forecasting. Zhang and Fang (2021) added Hurst index, a measure of volatility sustainability, to the model to improve the prediction performance, which confirmed the feasibility of predictor optimization. Generally speaking, trends is used to be measured by price indicators, including opening price, closing price, highest price and lowest price. And fluctuations are measured by volatility indicators, which can be classified into historical volatility, realized volatility and implied volatility based on different calculation methods. Because derivatives have the function of price discovery, a plethora of studies show that implied volatility has the highest correlation with price and can significantly improve the prediction accuracy (Day & Lewis, 1992; Fleming, 1998; Mayhew & Stivers, 2003). Derived from the Black–Scholes option pricing model (B–S model), implied volatility is more forward-looking than other indicators. In 1993, the Chicago Board Options Exchange (CBOE) compiled the Volatility Index (VIX) based on the implied volatility of options, which aims to measure the volatility of S&P500 index. And in 2003, the CBOE abandoned B–S model and created a model-free way to calculate the implied volatility. Onan et al. found that macroeconomic indicators, such as inflation rate and employment rate, have statistical correlations with VIX (Onan, Salih, & Yasar, 2014). And other researchers have shown that VIX can improve the accuracy of prediction (Matuozzo, Yoo, & Provetti, 2023; Pan, Wang, Liu, & Wang, 2019; Wang, Lin, Huang, & Wu, 2012; Xiao, Wen, Zhao, & Wang, 2021) and increase the return of quantitative investment strategy (de Almeida & Neves, 2022; Pinto, Neves, & Horta, 2015). In this paper, the VIX is calculated based on ETF options by a model-free way following CBOE.

Another way to optimize the model is to reduce the redundancy of input information. In 1950s, Bellman (1954) first proposed the term "curse of dimensionality" when study the theory of dynamic programming (Bellman, 1954). The "curse of dimensionality" refers to the difficulties in training machine learning models due to high-dimensional data, which is mainly reflected in two aspects: data sparsity and distance aggregation. In 1968, Gordon F. Hughes also found that when the number of training samples is fixed, the prediction ability of classifiers and regressors will decrease with the increase of data dimension (Hughes, 1968). This phenomenon was later called "Hughes phenomenon". In reality, in order to contain enough effective information, the training data is usually high-dimensional, which may have a negative impact on the generalization ability and prediction performance. Therefore, the method of dimensionality reduction, which is clarified as extracting features from original data (Zhu et al., 2013), has been widely studied and applied. Algorithms such as principal component analysis (PCA), genetic algorithm (GA), attention mechanism (AM) and Lasso, have been proved to significantly improve the prediction performance by filtering the noise of the data (Lin, Yang, & Song, 2009; Niu & Xu, 2020; Tsai & Hsiao, 2010; Vo, Jiawei, & Vo, 2013; Zhong & Enke, 2017). And among all these algorithms, Lasso has been proved to have outstanding filtering ability in most cases (Gao, Wang, & Zhou, 2021; Lv, Wang, Li, & Xiang, 2020; Rich, Livingston, & Morgan, 2020).

In general, this paper combines Lasso algorithm and GRU into a Lasso-GRU model for forecasting. Besides, based on the price discovery role of the option market, we construct the VIX as a representative of the implied volatility and employ it as a predictor. Fig. 1 has shown the prediction process. By conducting the model on the 5-min data of index futures, we find the new model outperforms the pure GRU and LSTM in training speed and accuracy on all three datasets. From the research, it brings new ideas for high-frequency traders to build strategies, for scholars to know more about the characteristics of high-frequency market. Besides, this paper also provide evidences for the key role of volatility in asset pricing, especially the implied volatility, indicating that it is crucial to study the investors' expectations. Furthermore, this paper also contributes to the application of data dimensionality reduction in prediction.

2. Methodology

2.1. Framework overview

As shown in Fig. 2, the model proposed in this article is divided into three parts: variable calculation, feature selection, and prediction.

(1) In the variable calculation section, in addition to collecting and preprocessing basic market data, this paper calculates the VIX index of the futures market under the guidance of the CBOE manual. There are seven original features, namely the opening price, closing price, highest price, lowest price, trading volume, turnover, and VIX.

(2) Before inputting variables into the prediction model, we use the Lasso algorithm to reduce dimensionality. This process compresses the coefficients of variables with insignificant contributions to 0, thereby reducing the number of parameters in the prediction model and improving prediction accuracy. The lower left corner of Fig. 2 shows the logic of Lasso feature selection using a two-dimensional variable as an example. In Section 3.2., the feature selection results of Lasso will be presented, with the turnover coefficient compressed to 0.

(3) In the prediction model section, we use GRU as the baseline, set three layers of GRU and convert the output dimension to 1 through the dense layer to obtain the predicted futures price. The upper part of Fig. 2 shows the overview of the prediction model, with the unit structure of GRU in the lower right corner.

2.2. Description of VIX

VIX is the abbreviation of volatility index, which is compiled based on the implied volatility of options. Because of the price discovery function of the option market, implied volatility usually represents the expectation of the future risk. Therefore, VIX is also known as the "panic index", which is used to reflect investors' estimates of the market trend in the coming month. The rise of VIX means that investors are pessimistic about the future market. Conversely, when VIX declines, it means that panic subsides and the market tends to be stable.

The calculation of VIX is based on options. To be specific, under the assumption that the asset price follows geometric Brownian motion and Ito's lemma, the asset price volatility can be solved, and then split through time integration. Following the derivation of Chow, Jiang, and Li (2021), VIX represents the expected value of future volatility, which can be calculated by the portfolio of out-of-the-money (OTM) options. This process is essentially a replication of risk neutral situation. A series of formulas for calculating VIX are shown in (1)–(5). The variables of subscript 1 are calculated on the recent month options while the variables of subscript 2 are based on the next recent month options.

$$VIX = 100 \times \sqrt{\{T_1\sigma_1^2[\frac{N_{T_2}-N_{30}}{N_{T_2}-N_{T_1}}] + T_2\sigma_1^2[\frac{N_{30}-N_{T_1}}{N_{T_2}-N_{T_1}}]\} \times \frac{N_{365}}{N_{30}}} \tag{1}$$

(i) σ_1^2 and σ_2^2 denote the implied volatility. The calculation formula is shown in (2).

$$\sigma^2 = \frac{2}{T}\sum_{i=1}\frac{\Delta k_i}{K_i^2}e^{RT}P(K_i) - \frac{1}{T}[\frac{F}{K_0}-1]^2 \tag{2}$$

In formula (2), F is the theoretical price of forward without arbitrage. K_0 is the strike price lower but closest to F, and $K_i(i = 1, 2, 3...)$ is the strike price of all front month options in ascending order. $P(K_i)$ is the price of the OTM option with the strike price of K_i (both call and put options are possible). Δk_i is the strike price interval corresponding

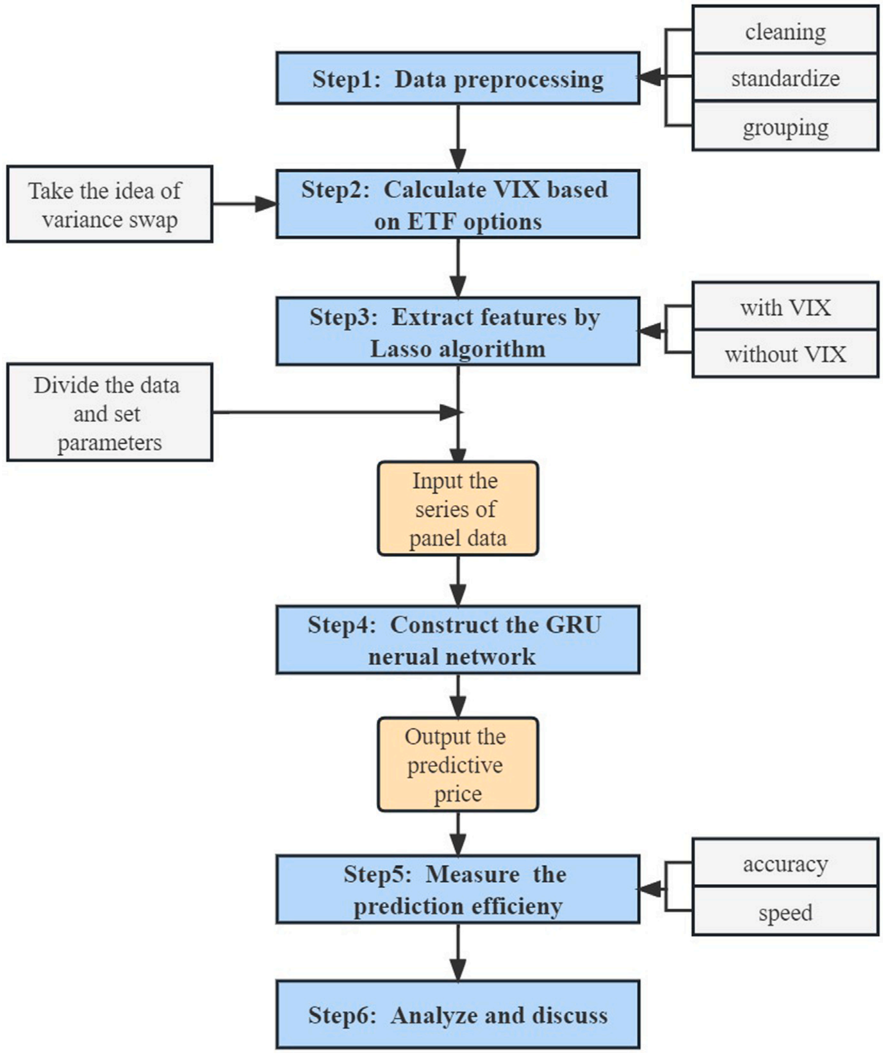

Fig. 1. The forecast process of VIX-Lasso-GRU model.

to the ith strike price, and R is the risk-free interest rate. Furthermore, F, $P(K_i)$ and Δk_i are defined respectively as follows. Compare the difference between the call option and put option prices in the same month, and recognize the option portfolio with the lowest absolute value of the difference as the at-the-money (ATM) option, which is the basic for calculating forward price. More specifically, all the following variables are based on the ATM option:

$$F = \text{strike price} + e^{RT} \times (\text{call option price} - \text{put option price}) \tag{3}$$

$$P(K_i) = \frac{C_i + P_i}{2} \tag{4}$$

$$\Delta k_i = \frac{K_{i+1} - K_{i-1}}{2} \tag{5}$$

(ii) T_1 and T_2 stand for the residual maturity measured in years while N_{T_1} and N_{T_2} are representative of the residual maturity measured in days. N_{365} and N_{30} denote 365 days and 30 days separately.

2.3. Lasso algorithm

Lasso algorithm was first proposed by Tibshirani (1996). It is a biased estimation method based on the linear model. Its principle is to construct a penalty function to compress the regression coefficients. In this process, the coefficients of unimportant variables are compressed to 0, which is also understood as the process of reducing the dimension.

Lasso algorithm has many advantages. First of all, it has a high tolerance for data. Specifically, it can deal with nonnegative exponential variables, multi-dimensional continuous variables and multi-dimensional discrete variables. Secondly, it can be used to filter variables, which is picking out more influential variables while removing unimportant or irrelevant variables. It is important to determine the number of variables in a model. Too many variables may lead to over-fitting, while too few variables will reduce the goodness of fit. Lasso algorithm not only helps to reduce the complexity of the model and avoid over-fitting problems, but also helps to improve the fitting accuracy.

The principle of lasso algorithm is introduced in detail, a linear regression model is shown as formula (6).

$$Y = X\beta + \epsilon \tag{6}$$

Y is an $n{\times}1$ vector and $X = (X_1, X_2, \ldots, X_p)$ is an $n{\times}p$ matrix. Shaped as $p \times 1$, β is a parameter vector to be estimated. ϵ represents random error, assuming a normal distribution. The parameters are estimated by the least square method to minimize the random error. Set $F(\beta)$ be the objective function, which is defined as (7).

$$F(\beta) = \|\epsilon\|^2 = \|Y - X\beta\|^2 = (Y - X\beta)(Y - X\beta)' \tag{7}$$

When $F(\beta)$ gets minimized, the solution is obtained as below.

$$\hat{\beta} = (X'X)^{-1}X'Y \tag{8}$$

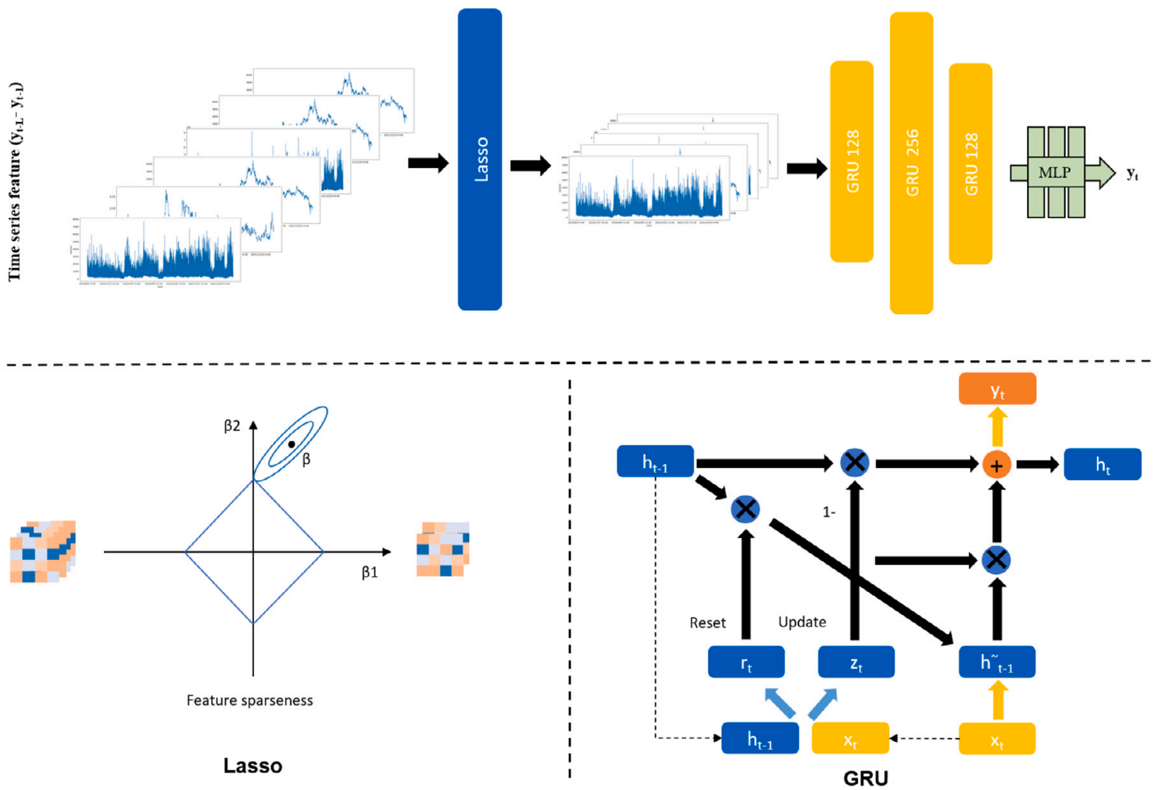

Fig. 2. The framework of VIX-Lasso-GRU model.

The problem of linear regression model is that when there is multicollinearity between variables, the rank of variable matrix X is less than p. Thus it is impossible to obtain an unbiased estimation of β. However, this problem can be solved by ridge regression, which adds penalties for β compared with the linear model. Then the objective function $F(\beta)$ changed into:

$$F(\beta) = \|Y - X\beta\|^2 + \lambda \|\beta\|_2^2 \tag{9}$$

And the solution is here, where $\lambda \geq 0$:

$$\hat{\beta}_{\text{ridge}} = (X^T X + \lambda I)^{-1} X^T Y \tag{10}$$

According to the above formula, λ guarantees that $(X^T X + \lambda I)^{-1}$ is full rank and reversible. And its value also determines the penalty strength. The greater the value of λ, the stronger the penalty. However, λ does not tend to infinity, which makes ridge regression unable to compress the regression coefficient to 0. In other words, those insignificant variables are still left to be estimated. Different from ridge regression, Lasso changes the 2-norm of β into 1-norm in the objective function, so that the objective function is

$$F(\beta) = \|Y - X\beta\|^2 + \lambda \|\beta\|_1 \tag{11}$$

When the constraint on β becomes 1-norm (the sum of absolute values of each element), the regression coefficient can be compressed to zero, so as to truly achieve the purpose of dimension reduction. The superiority of lasso algorithm also makes it widely used in economic and financial fields (Fang, Sun, Li, & Yu, 2018; He, Qin, Wang, Wang, & Wang, 2019; Huebner, Hamilton, Chalabi, Shipworth, & Oreszczyn, 2015; Panagiotidis, Stengos, & Vravosinos, 2018).

2.4. GRU neural network

Neural network is essentially a mathematical model. It is named because its internal structure and node connection are similar to the synapse of human brain. Based on the network topology, neural network is divided into three parts: input layer, hidden layer and output layer. Different types of neurons form corresponding network layers, which are connected with each other. Input neuron is responsible for receiving external information. Hidden neuron and output neuron are responsible for information processing and output results respectively. Neural network has the characteristics of self-learning and is good at dealing with complex nonlinear problems.

Neural network has various forms, which can meet the needs of different types of data. Numerous studies have shown that financial time series have long-memory feature (Al-Yahyaee, Mensi, & Yoon, 2018; Bentes, Menezes, & Mendes, 2008; Ding & Granger, 1996; Ding, Granger, & Engle, 1993). GRU has two gates: update gate and reset gate, which are used to determine what historical information will be saved and output. It is such gated structure that endows the model with long-term memory, which works for transferring historical information and dropping noise to simplify the model. The structure of GRU is shown in Fig. 3.

The update gate controls how much information in the historical output h_{t-1} is contained in the current time point output h_t, and how much effective information $\tilde{h}_t$ at the current time point is saved. The formula of update gate is as (12). σ is an activation function, such as sigmoid and tanh. x_t and h_{t-1} stand for the current input value and the previous output respectively. w_z and v_z are the weight functions of the corresponding value. θ_z is a bias term.

$$z_t = \sigma(w_z x_t + v_z h_{t-1} + \theta_z) \tag{12}$$

The reset gate controls the influence degree of the historical information h_{t-1} on the effective information $\tilde{h}_t$ stored at the current time. When the two are independent, the output of the gate is 0. The formula of reset gate is exactly the same as that of update gate formally.

$$r_t = \sigma(w_r x_t + v_r h_{t-1} + \theta_r) \tag{13}$$

In the formula, the definitions of $\tilde{h}_t$ and h_t are as below.

$$\tilde{h}_t = \tanh(w_c x_t + v_c(r_t * h_{t-1}) + \theta_c) \tag{14}$$

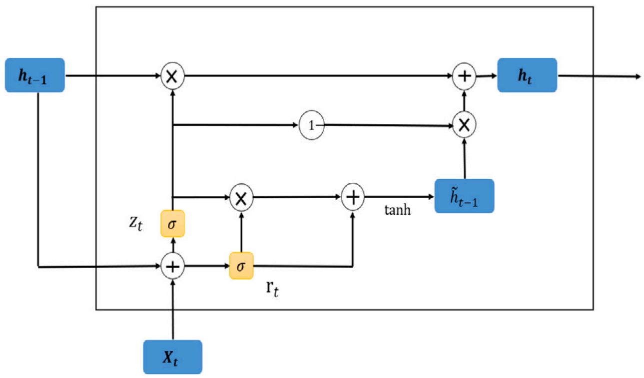

Fig. 3. The structure of a cell in GRU: Marked in light yellow, z_t and r_t are gate controls, representing the update gate and the reset gate respectively. Besides, input and output variables are highlighted in blue.

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \tag{15}$$

When $z_t = 0$ and $r_t = 1$, GRU works as RNN. The update gate and reset gate enable effective historical signals to be continuously transmitted in GRU, which solve the problem of gradient disappearance in RNN.

3. Experiment preparation

3.1. Data

3.1.1. Dataset description

We aims to apply our new model on prediction of the 5-min intraday data of index futures. We conducted experiments on three sets of stock index futures data to ensure the robustness of the model. The three datasets are SSE 50 index futures, CSI 300 index futures, and CSI 500 index futures. We selected experimental intervals based on the listing time, so the ranges of the three datasets are different. While ensuring sufficient data volume for model training, the intervals include different market forms such as bull and bear markets, which also tests the predictive efficiency of the model.

(1) The underlying asset of SSE 50 index futures is SSE 50 index, which covers 50 stocks with large market cap and good liquidity in Shanghai Stock Exchange, comprehensively reflecting the overall performance of a group of leading enterprises with the most market influence in the Shanghai securities market. The options used to calculate the VIX index are the SSE 50 ETF options. The range of the dataset is from June 3, 2019 to March 30, 2022.

(2) With the CSI 300 Index as the underlying asset, the CSI 300 index futures were launched by the China Financial Futures Exchange on April 16, 2010, which is also China's first stock index futures contract. The CSI 300 Index selects 300 large and liquid stocks from the Shanghai and Shenzhen stock markets as samples, comprehensively reflecting the overall performance of listed stock prices in the Chinese market. And the option data selects CSI 300 ETF options listed on SSE. The sample is from December 23, 2019 to March 30, 2023.

(3) The CSI 500 stock index futures are based on the CSI 500 index as the underlying asset, which reflects the overall situation of small and medium-sized companies in the Shanghai and Shenzhen securities markets. Therefore, this dataset is an effective supplement to the two datasets reflecting large market capitalization stocks mentioned above. The ETF options corresponding to the CSI 500 Index were listed on September 19, 2022, and we used this as the starting point for the sample until September 1, 2023.

Table 1
The partition of datasets.

Dataset	Training set	Valid set	Test set	Total
SSE 50	26 379	3297	3298	32 974
CSI 300	34 444	4305	4306	43 055
CSI 500	8908	1113	1114	11 135

All three datasets are from the JoinQuant Platform.[1] Futures data selects 5-min data from consecutive main contracts, including opening price, closing price, highest price, lowest price, trading volume, and turnover. The option data adopts ETF options corresponding to the index. The characteristics of the datasets are shown in the Table 1.

3.1.2. Data preprocessing

(1) Variable calculation section: In order to avoid the impact of different dimensions on model training, we standardizes the data before input by the Formula (16).

$$x_i' = \frac{x_i - min(x_i)}{max(x_i) - min(x_i)} \tag{16}$$

In addition, for the missing value, the previous nonempty data is used to fill in. The calculation of VIX refers to the researches of Bakshi, Charles, and Chen (1997) and Zhang and Xiang (2008), and two types of options are excluded. (i) Remove options with a remaining maturity of less than 7 days to avoid liquidity bias near due date. (ii) Remove options with a closing price of less than 2 yuan (the SSE charges a commission of 2RMB per contract) to mitigate the impact of price dispersion. (2) Model training section: Based on the experience of related researches, this paper divides the datasets into "training set: validation set: test set", with a ratio of 8:1:1.

3.1.3. VIX calculation

The data for calculating VIX involve the implied volatility of options and risk-free interest rates. The implied volatility is calculated by the model-free method. We take Shanghai Interbank Offered Rate (Shibor) as the risk-free rate, and the complete term structure of interest rate is obtained by linear interpolating.

VIX is calculated on each trading day. First of all, we filter out the front month and sub-front month options and rank them in ascending order of exercise price respectively. Then, for each group, calculate the differences between prices of call options and put options with the same exercise price. ATM options is chosen as the least absolute value of differences. On the basis of ATM options, we pick out the OTM options with different exercise prices, and sum up the variance contributions. At the same time, calculate the term structure of interest rates and a series of other indicators, which have been shown in the methodology part.

The time series of VIX is shown in Fig. 4. It can be clearly seen that there is an upsurge of VIX at the beginning of 2020, due to the outbreak of COVID-19. Besides, a series of ups and downs are caused by complex situations at home and abroad. And we need to convert the daily VIX into a 5-min sequence before adding it to input list. In this paper, the daily VIX data is corresponding to the first 5-min data of the day, and then the forward push method is used to fill the gap.

3.2. Dimensionality reduction by Lasso

In this section, Lasso algorithm is used to reduce the dimension of input factors, that is, to extract features. The initial factors selected in this paper include opening price, closing price, highest price, lowest price, trading volume, turnover, and VIX. As mentioned earlier, factors

[1] https://www.joinquant.com/

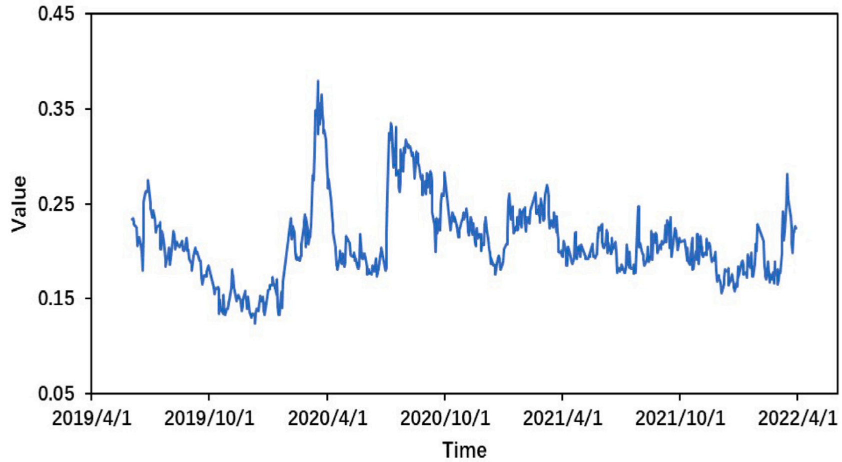

Fig. 4. Time series of VIX calculated based on SSE 50 ETF options.

that measure trends and fluctuations should be both taken into account. Here is a brief theoretical analysis of why we choose these features.

(i) The opening price and closing price are the basis for the calculation of many technical indicators, such as the moving average and MACD, which can reflect the trend of assets. In addition, the difference between the opening price and the closing price also roughly reflects the volatility of the certain period.

(ii) The highest price and the lowest price are the results of the game between buyers and sellers, reflecting market sentiment. If the difference between the opening price and the closing price focuses on the result of volatility, the difference between the highest price and the lowest price mainly reflects the process to get such result.

(iii) Trading volume and turnover are important indicators of volume price analysis (VPA) proposed by Richard Wyckoff, a pioneer in the field. Zhang, Ding, and Scheffel (2018) found that the factors driving short-term volatility, micro supply and demand changes, also drive trading volume and liquidity. And here we choose turnover as a measure of liquidity.

(iv) As a proxy of implied volatility, VIX not only contains volatility information, but also reflects investors' expectations for the future market. Calculating on the basis of options, it is taken for granted that VIX represents the implied volatility of the SSE 50 index directly. But studies have shown that there is a natural link and spillover effect between futures and spot (Magkonis & Tsouknidis, 2017; Miljkovic & Goetz, 2020), which makes the changes of two markets interrelated. Therefore it is reasonable to employ VIX for predicting futures price.

In order to test the effectiveness of VIX, we use lasso algorithm to extract the effective factors before and after adding VIX. Lasso algorithm adds penalty terms to the linear regression model to achieve the purpose of extracting features. There are two ways to select the penalty parameter λ. The first method is to subjectively select λ when the regression coefficient is the most stable by observing the image, which is shown as Fig. 5. From Fig. 5, we can see that for several variables with relatively small contributions, such as trading volume, turnover, and VIX, it is difficult for us to subjectively distinguish the weights assigned to them by Lasso. However, Lasso requires a very careful parameter selection. So here we turn to the other method, which is to select λ with the smallest regression error through the LassoCV function. The LassoCV function uses cross validation to evaluate the optimal λ by minimizing mean square error (MSE). We divided the training and validation sets into 9:1. After several iterations, the optimal λ is 3.47e−08 when the VIX index is not added, and 3.18e−08 when the VIX index is added. (The best λs have been marked as the vertical dashed lines in Fig. 5).

Fig. 6 and Table 2 show the selection results when the penalty parameter λ equals the best value calculated by LassoCV strategy. Three conclusions can be drawn from the results.

(i) Both sets of variables show that the weights of price indicators are larger than those of volume indicators. And the closing price is given the greatest coefficient, indicating that it is highly correlated with the price at the next moment, which is consistent with our previous theoretical analysis.

(ii) The coefficients of trading volume and turnover are compressed close to 0, which exceeded our expectations. Especially after the addition of VIX, turnover does not play a role with coefficient of 0.

(iii) The absolute coefficient of VIX is significantly non-zero, indicating that it is effective for predicting prices. And the negative coefficient indicates a negative correlation with future price. This can be explained by the fact, which is that a rise in VIX indicates a surge in panic, usually accompanied by a subsequent decline in the market. In addition, from the data in Table 2, we can see that the addition of VIX reduces the coefficient of opening price, closing price and turnover. We believe that VIX has a partial substitution effect.

4. Experiments

4.1. Baseline models

Guided by the feature extraction results of Lasso algorithm, it can be seen that the turnover has the smallest weight among all variables, especially when VIX is introduced. Therefore, it is regarded as a redundant variable in this paper. Then we set four sets of input features as follows. Model 2 and 4 are compared with Model 1 and 3 respectively to test the effectiveness of lasso algorithm in extracting features. Model 1 and 2 are compared with Model 3 and 4 respectively to prove whether the introduction of VIX helps to improve forecast performance.

Model 1: opening price, closing price, highest price, lowest price, trading volume, turnover

Model 2: opening price, closing price, highest price, lowest price, trading volume

Model 3: opening price, closing price, highest price, lowest price, trading volume, turnover, VIX

Model 4: opening price, closing price, highest price, lowest price, trading volume, VIX

In order to better evaluate the predictive performance of the model proposed in this article, we replaced GRU with LSTM for comparative experiments. The prediction model is divided into four layers, with the first three layers being the GRU (LSTM) layer, of which the optimal number of hidden layers (units) is 128/256/128 respectively. After each layer of GRU (LSTM), a dropout layer is added to prevent overfitting, and the dropout rate is set to 0.2. Finally, the output dimension is converted to 1 through a dense layer, which outputs the predicted closing price.

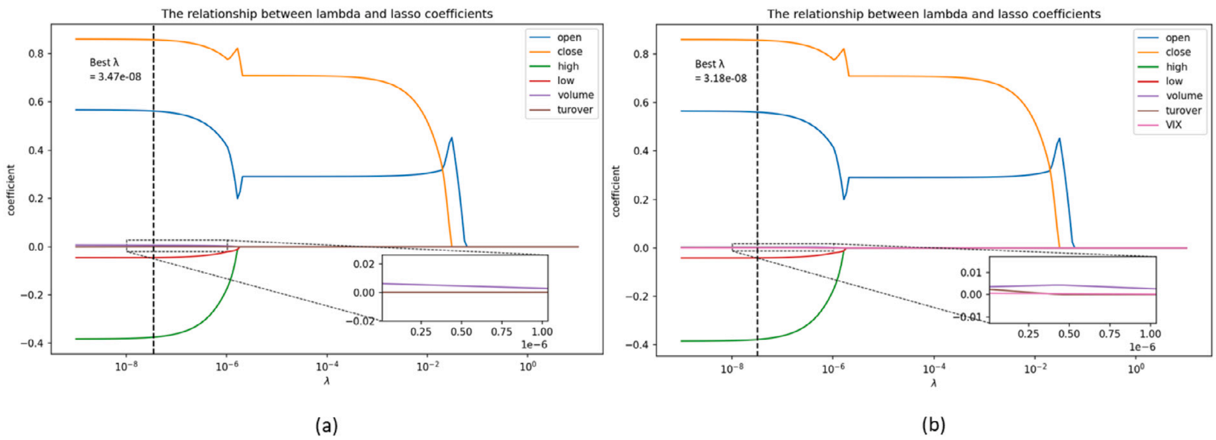

Fig. 5. Coefficients of variables under different penalty parameter λ. The (a) shows the selection of variables by Lasso under different λ without adding the VIX index. And the (b) shows the variation of the coefficients of each feature with the λ after adding the VIX index.

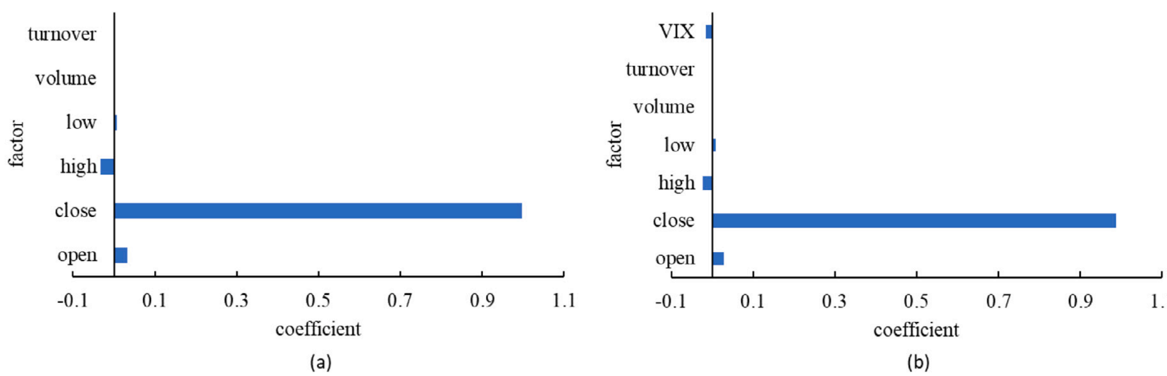

Fig. 6. The coefficients of each factor fitted by Lasso algorithm: without VIX (a) and with VIX (b).

Table 2
Lasso algorithm feature extraction results.

Set	Open	Close	High	Low	Volume	Turnover	VIX
Without VIX	0.03288	0.99703	−0.03392	0.00781	0.00138	0.00007	–
With VIX	0.02924	0.98642	−0.02392	0.00804	0.00160	0.00000	−0.01533

We adopt a one-step forward prediction method. By setting a sliding window, data from the past period is input to predict the closing price at the current time. Including the sliding window length, the model also includes other hyperparameters, and we determine the values through grid search. In the three data sets, all hyperparameters are consistent except for epoch and sliding window length. After grid search, we sets the optimal hyperparameter combination as follows: batch size of 256, epoch of 100/120/140 for SSE 50, CSI 300 & CSI 500 dataset respectively, sliding window length of 10/8/12 for SSE 50, CSI 300 & CSI 500 dataset respectively. The optimal hyperparameter setting is based on the experimental results of GRU baseline.

4.2. Analysis of prediction results

We measure the predictive efficiency of the model from two dimensions: accuracy and training speed. Select three popular indicators for accuracy, including MAE, MSE (also the loss function of the GRU model), and R^2. The formulas of these metrics are listed as below. The training speed is evaluated by the training time and the number of model parameters.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|^2 \tag{17}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{18}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{19}$$

Next, we will report in detail the experimental results under three datasets, including the trained loss function, prediction graph, and evaluation metrics. Due to the similarity in the experimental results among the three datasets, this article focuses on the SSE 50 dataset for explanation, and validates it with the results of the other two datasets. Fig. 7 shows the loss function curves for SSE 50 prediction under GRU baseline. It can be seen that the loss of training set and validation set decreases steadily with the fitting process, and converges to a lower loss level in the end. This indicates that the models are feasible, without over-fitting and under-fitting.

After the model training, the output data is de-standardized and the forecast price is drawn in Figs. 8 and 9. From the graphs, it can be seen that Model 1 (a) has the worst fitting performance, while Model 4 (d) has the best fitting performance. In the composite graph of the four models, we intentionally zoomed in the range of market with higher volatility to compare the accuracy of the four models in predicting

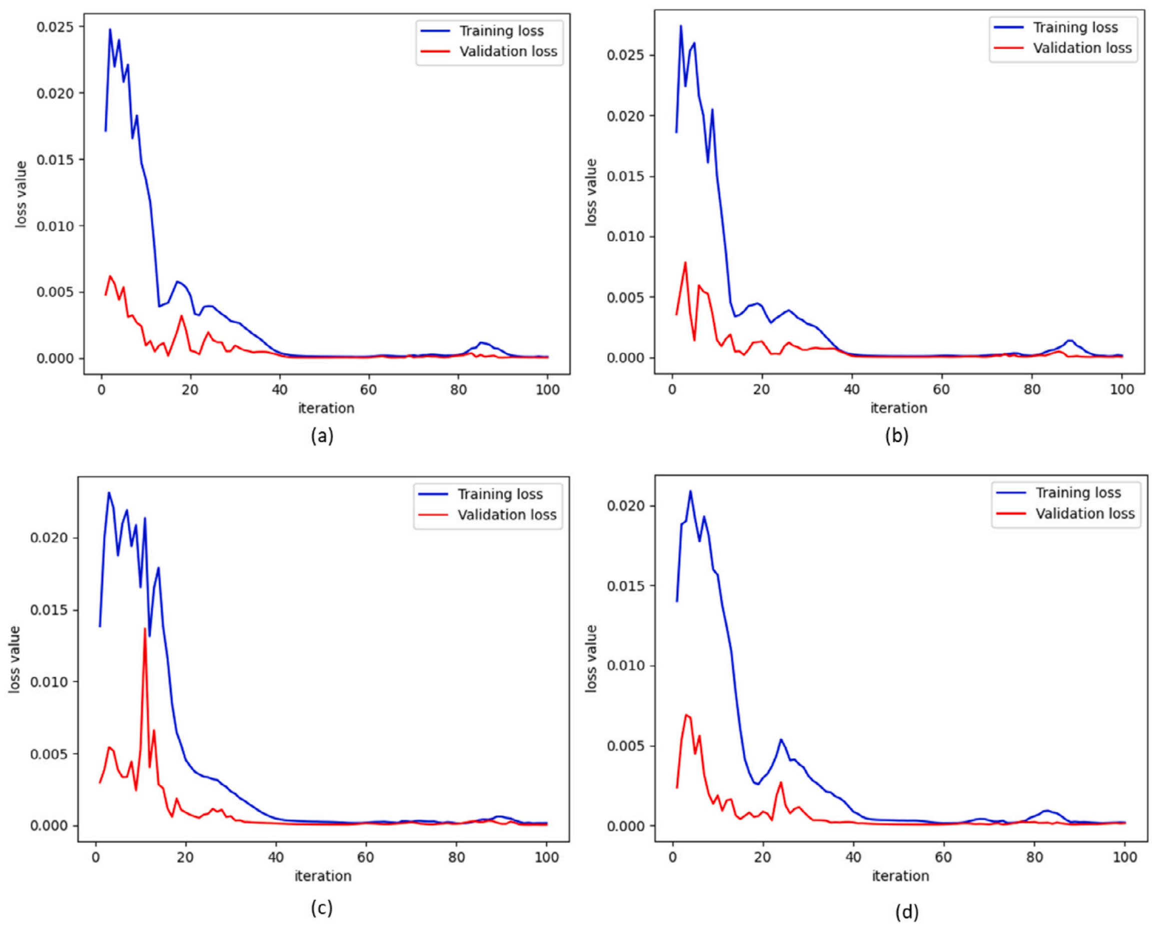

Fig. 7. Loss curve for SSE 50 prediction under GRU baseline. (a)–(d) represent the results of Model 1–4 respectively.

extreme situations. It can be seen that the predicted values of Model 4 represented by the red curve still best fit the actual values, and it is preliminarily judged that the input of the VIX index can effectively improve the prediction accuracy.

Table 3 shows the experimental results of three datasets. As mentioned in Section 4.1, the setting of hyperparameters is determined based on grid search of GRU baseline. In order to compare the effectiveness of GRU and LSTM, this article replaces GRU with LSTM under the same parameters. However, during the experimental phase, we found that when predicting the SSE 50 and CSI300 data sets, the LSTM model did not converge, so it was not reported in the table. This also reflects that LSTM has stricter requirements for parameter settings. The bold data in the table distinguishes the best among the same group.

(1) Accuracy

Although the combination of optimal hyperparameters varies, the experimental results show that Model 4, the combination with the addition of VIX index and compression of turnover to 0, performs best when using GRU as the baseline, which means the prediction error is the smallest.

In the CSI 500 dataset, comparing the performance results using GRU and LSTM as the baseline, it was found that the prediction accuracy of LSTM is generally higher (the prediction errors of LSTM in models 1, 2, and 3 are smaller than those of GRU), which is consistent with the expected results. However, this article also found that under the LSTM baseline, the errors of the four sets of inputs were not significantly different, that is, the marginal error reduction caused by the input of additional useful information was small. In contrast, the variance of GRU prediction error is large. Taking Model 4 as an example, it is found that using GRU as the baseline yields higher accuracy.

Furthermore, comparing Model 1 and Model 2, we found that the addition of turnover did not significantly improve model accuracy, and even increased prediction error (SSE 50 and CSI 300 datasets). This also indicates the necessity of using Lasso algorithm for feature selection before inputting the model.

(2) Training speed

The comparison of training time and model parameters reflects the use of Lasso algorithm in simplifying models and improving training speed.

Comparing Model 2 with Model 3, which shows a difference of 2 in the number of input variables, it can be observed that the training time has significantly increased by more than 10%. At the same time, under the established model structure in this article, the number of trainable parameters also increases with the increase of input variables. Although the increase proportion is only about 2%, the number of parameters will rapidly increase with the complexity of the network structure and the increase of hidden layers.

Another noteworthy point is that comparing the training time and parameter size of GRU and LSTM models under the CSI 500 dataset, it was found that GRU has an absolute speed advantage. Taking Model 4 as an example, in achieving higher accuracy, the training time of the GRU model is 266 s shorter than that of LSTM, accounting for nearly 50% of the total GRU time. For high-frequency financial data, fast and accurate prediction is the key to constructing quantitative strategies, and even prediction speed is more important than accuracy. Because under the guidance of the law of large numbers, a successful strategy only requires a winning rate above 50%, it is wise to choose a model with faster prediction speed when both models meet the winning rate requirements.

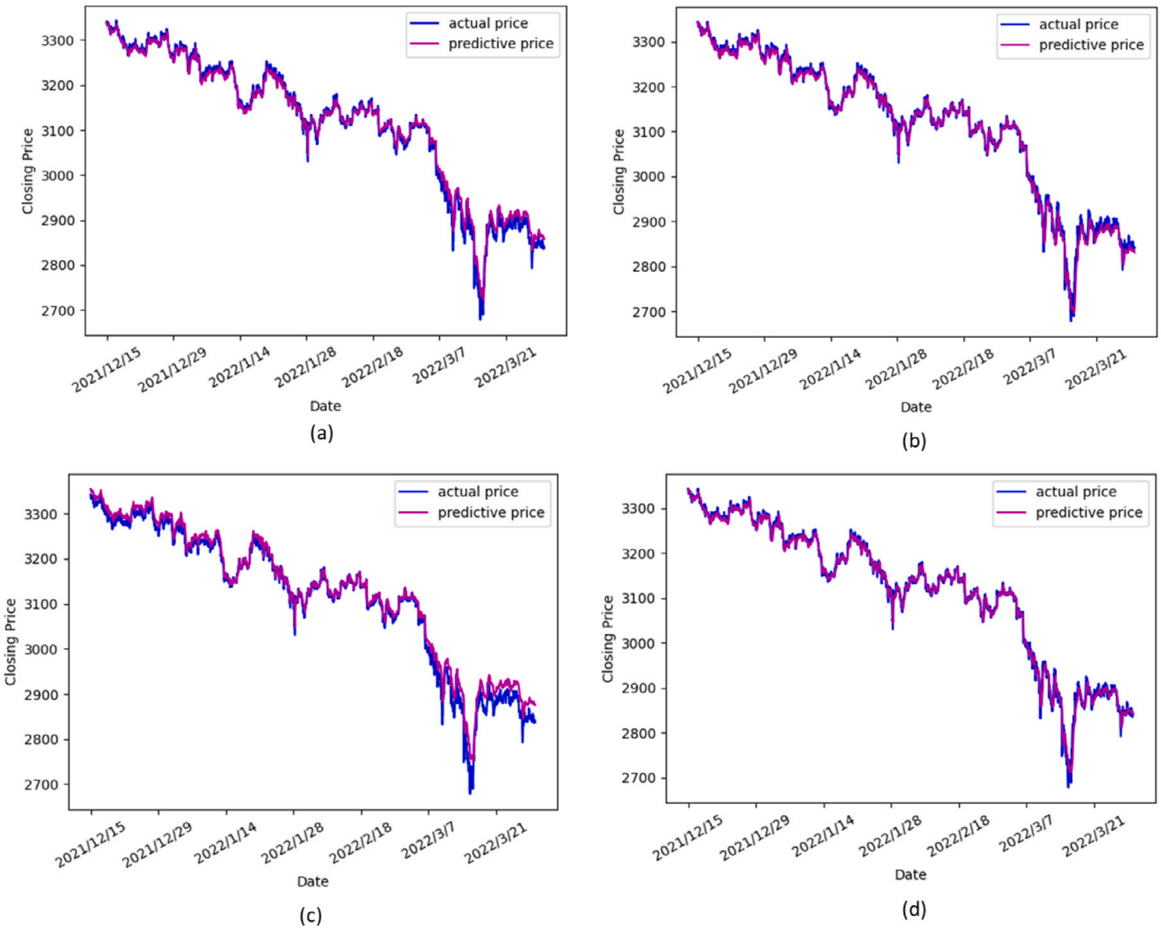

Fig. 8. Prediction of SSE 50 under GRU baseline. (a)–(d) represent the predictions by Model 1–4 respectively.

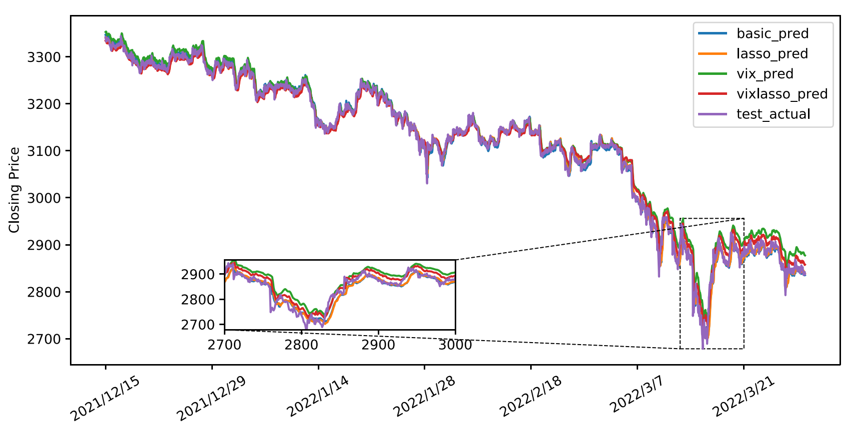

Fig. 9. Prediction of SSE 50 under GRU baseline with a zoomed-in subgraph during extreme situation. Legends correspond to Model 1–4 from top to bottom, and the line of "test_actual" depicts the real price.

Finally, we conducted ablation study to investigate the effect of sliding window length on experimental results, using the SSE 50 dataset as an example. Window length means how long historical information predicts future prices. If the window is too long, it may contain useless information and reduce the training speed. If the window is too short, there may be issues of missing information and low prediction accuracy. From Table 4, it can be seen that as the length of the sliding window increases, the prediction error shows a trend of first decreasing and then increasing, while the prediction time show the same trend.

5. Conclusions and discussion

A large number of scholars and practitioners are engaged in the construction of price prediction models, which is of great significance to increase returns and avoid risks. Compared with developed countries such as the America and Japan, the financial markets are less efficient in China, making price forecasts under great attention. Especially for stock index futures market, its short-term mechanism and $T+0$ trading system provide great potential for high-frequency strategies. Therefore,

Table 3
Prediction results based on GRU and LSTM in three datasets.

Models		SSE 50 index futures (window length = 10)				
		MSE	MAE	R^2	Time	Params
GRU	Model 1	314.5820	12.9200	0.9870	961 s	497 025
	Model 2	255.6643	11.0642	0.9890	991 s	**496 641**
	Model 3	515.1050	19.3710	0.9780	967 s	497 409
	Model 4	**138.0030**	**7.8060**	**0.9940**	976 s	497 025
Models		CSI 300 index futures (window length = 8)				
		MSE	MAE	R^2	Time	Params
GRU	Model 1	404.0044	15.5102	0.9440	1225 s	497 025
	Model 2	278.6635	12.7298	0.9614	1205 s	**496 641**
	Model 3	395.7864	15.7865	0.9451	1308 s	497 409
	Model 4	**227.2289**	**10.6824**	**0.9685**	1222 s	497 025
Models		CSI 500 index futures (window length = 12)				
		MSE	MAE	R^2	Time	Params
GRU	Model 1	780.4793	22.3139	0.9700	536 s	497 025
	Model 2	1806.4559	34.3472	0.9307	551 s	**496 641**
	Model 3	620.5967	18.1293	0.9762	732 s	497 409
	Model 4	**279.2154**	**11.2076**	**0.9893**	534 s	497 025
LSTM	Model 1	436.3153	15.2552	0.9833	897 s	1 320 193
	Model 2	575.7178	17.5278	0.9780	840 s	**1 319 169**
	Model 3	472.5579	15.9902	0.9820	918 s	1 321 217
	Model 4	**424.9455**	**15.2030**	**0.9837**	900 s	1 320 193

Table 4
Prediction of SSE 50 under GRU baseline with different window length.

Metrics		MSE	MAE	R^2	Time
Window length	$T = 5$	309.2523	12.9623	0.9867	1319 s
	$T = 10$	138.0030	7.8060	0.9940	991 s
	$T = 15$	1607.6692	31.3749	0.9309	2976 s
	$T = 20$	1610.9959	29.6667	0.9307	3653 s

this paper attempts to build a better prediction model called VIX-Lasso-GRU Model which is based on the existing research and proves its effectiveness on the 5-min data of three index futures, namely SSE 50, CSI 300 & CSI 500. Based on pure GRU neural network, we optimize it in two ways. One is to calculate VIX as a proxy variable of implied volatility by ETF options and treat it as an input factor. The other way is to extract features of the input factors through Lasso algorithm to reduce data redundancy. In addition to VIX, the factors to be selected include opening price, closing price, highest price, lowest price, trading volume and turnover. Before predicting, we preprocess the data and get two conclusions. (i) After regression by Lasso algorithm, the coefficient of turnover is compressed almost to 0. So it is removed from the input factors list. (ii) The regression coefficient of VIX is significantly not 0, and its addition reduces the coefficient of other factors, indicating that VIX contains new information. According to the results, we construct four models of different input factors to test the effectiveness of VIX and Lasso algorithms. It turns out that the addition of VIX significantly improves the prediction accuracy, while Lasso algorithm can promote the training process by selecting useful features and lowering time cost. Also, we make a comparison between GRU and LSTM baseline, and the results improve that GRU maintains high prediction accuracy while saving a significant amount of time cost.

In this paper, we prove the feasibility of VIX and Lasso algorithm in optimizing prediction model. The purpose of this article is not only to provide a better model, but also to bring inspirations to stakeholders. For investors, they always try to reduce risks and obtain higher returns. So our new model provides a prototype for high-frequency trading strategies which is based on prediction. For scholars, VIX is helpful to get a deeper understanding of the relationship between derivatives and their underlying assets. It can also be inferred that the volatility does play a key role in asset pricing, so it of great significance to predict volatility, and VIX has set a good example. For policymakers, VIX has proved to be a predictor of market sentiment. It is necessary to compile volatility indices in all futures markets, which helps to detect market expectations in a timely manner and intervene effectively when necessary.

Finally, we also put forward prospects for future work. First, this paper verifies the possibility of improving the efficiency of problem solving under high-dimensional data. With the help of data mining, we can get more heterogeneous factors, which may be helpful for prediction and pricing. Second, we prefer to use simpler and more effective models with great explainability.

CRediT authorship contribution statement

Wen Fang: Conceptualization, Methodology, Supervision, Project administration, Writing – original draft, Writing – review & editing, Funding acquisition. **Shuwen Zhang:** Data curation, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Visualization, Writing – original draft, Writing – review & editing. **Chang Xu:** Methodology, Software, Validation, Formal analysis, Investigation, Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

The authors are supported by National Natural Science Foundation of China [grant number 72101020].

References

Al-Yahyaee, K., Mensi, W., & Yoon, S. (2018). Efficiency, multifractality, and the long-memory property of the bitcoin market: A comparative analysis with stock, currency, and gold markets. *Finance Research Letters*, *27*, 228–234. http://dx.doi.org/10.1016/j.frl.2018.03.017.

Bakshi, G., Charles, C., & Chen, Z. (1997). Empirical performance of alternative option pricing models. *The Journal of Finance*, *52*(5), 2003–2049. http://dx.doi.org/10.1111/j.1540-6261.1997.tb02749.x.

Bellman, R. (1954). The theory of dynamic programming. *American Mathematical Society. Bulletin*, *60*(6), 503–515. http://dx.doi.org/10.1090/S0002-9904-1954-09848-8.

Bentes, S., Menezes, R., & Mendes, D. (2008). Long memory and volatility clustering: Is the empirical evidence consistent across stock markets? *Physica A. Statistical Mechanics and its Applications*, *387*(15), 3826–3830.

Bhardwaj, G., & Swanson, N. (2006). An empirical investigation of the usefulness of ARFIMA models for predicting macroeconomic and financial time series. *Journal of Econometrics*, *131*(1–2), 539–578. http://dx.doi.org/10.1016/j.jeconom.2005.01.016.

Cao, J., Li, Z., & Li, J. (2019). Financial time series forecasting model based on CEEMDAN and LSTM. *Physica A: Statistical Mechanics and its Applications*, *519*, 127–139. http://dx.doi.org/10.1016/j.physa.2018.11.061.

Chen, R., & Pan, B. (2016). Chinese stock index futures price fluctuation analysis and prediction based on complementary ensemble empirical mode decomposition. *Mathematical Problems in Engineering*, *2016*.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *EMNLP 2014-2014 conference on empirical methods in natural language processing, proceedings of the conference* (pp. 1724–1734). Association for Computational Linguistics (ACL), http://dx.doi.org/10.3115/v1/d14-1179.

Chow, K., Jiang, W., & Li, J. (2021). Does VIX truly measure return volatility? In *Handbook of financial econometrics, mathematics, statistics, and machine learning* (pp. 1533–1559).

Day, T. E., & Lewis, C. M. (1992). Stock market volatility and the information content of stock index options. *Journal of Econometrics*, *52*(1-2), 267–287.

de Almeida, R., & Neves, R. (2022). Stock market prediction and portfolio composition using a hybrid approach combined with self-adaptive evolutionary algorithm. *Expert Systems with Applications*, *204*, Article 117478. http://dx.doi.org/10.1016/j.eswa.2022.117478.

Ding, S., Cui, T., Xiong, X., & Bai, R. (2020). Forecasting stock market return with nonlinearity: a genetic programming approach. *Journal of Ambient Intelligence and Humanized Computing*, *11*(11), 4927–4939. http://dx.doi.org/10.1007/s12652-020-01762-0.

Ding, Z., & Granger, C. (1996). Modeling volatility persistence of speculative returns: A new approach. *Journal of Econometrics*, *73*(1), 185–215. http://dx.doi.org/10.1016/0304-4076(95)01737-2.

Ding, Z., Granger, C., & Engle, R. (1993). A long memory property of stock market returns and a new model. *Journal of Empirical Finance*, *1*(1), 83–106. http://dx.doi.org/10.1016/0927-5398(93)90006-D.

Fang, L., Sun, B., Li, H., & Yu, H. (2018). Systemic risk network of Chinese financial institutions. *Emerging Markets Review*, *35*, 190–206. http://dx.doi.org/10.1016/j.ememar.2018.02.003.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, *270*(2), 654–669. http://dx.doi.org/10.1016/j.ejor.2017.11.054.

Fleming, J. (1998). The quality of market volatility forecasts implied by S & P 100 index option prices. *Journal of Empirical Finance*, *5*(4), 317–345. http://dx.doi.org/10.1016/S0927-5398(98)00002-4.

Gao, Y., Wang, R., & Zhou, E. (2021). Stock prediction based on optimized LSTM and GRU models. *Scientific Programming*, (2021), http://dx.doi.org/10.1155/2021/4055281.

He, Y., Qin, Y., Wang, S., Wang, X., & Wang, C. (2019). Electricity consumption probability density forecasting method based on LASSO-quantile regression neural network. *Applied Energy*, *233–234*, 565–575. http://dx.doi.org/10.1016/j.apenergy.2018.10.061.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. http://dx.doi.org/10.1162/neco.1997.9.8.1735.

Huebner, G., Hamilton, I., Chalabi, Z., Shipworth, D., & Oreszczyn, T. (2015). Explaining domestic energy consumption–the comparative contribution of building factors, socio-demographics, behaviours and attitudes. *Applied Energy*, *159*, 589–600.

Hughes, G. (1968). On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, *14*(1), 55–63. http://dx.doi.org/10.1109/TIT.1968.1054102.

Jiang, M., Liu, J., Zhang, L., & Liu, C. (2020). An improved stacking framework for stock index prediction by leveraging tree-based ensemble models and deep learning algorithms. *Physica A: Statistical Mechanics and its Applications*, (541), http://dx.doi.org/10.1016/j.physa.2019.122272.

Kim, H., & Won, C. (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. http://dx.doi.org/10.1016/j.eswa.2018.03.002.

Li, Y., Shen, D., Wang, P., & Zhang, W. (2020). Does intraday time-series momentum exist in Chinese stock index futures market? *Finance Research Letters*, (35), http://dx.doi.org/10.1016/j.frl.2019.09.007.

Li, X., Xie, H., Wang, R., Cai, Y., Cao, J., Wang, F., . . ., & Deng, X. (2016). Empirical analysis: stock market prediction via extreme learning machine. *Neural Computing and Applications*, *27*(1), 67–78. http://dx.doi.org/10.1007/s00521-014-1550-z.

Lin, X., Yang, Z., & Song, Y. (2009). Short-term stock price prediction based on echo state networks. *Expert Systems with Applications*, *36*(3), 7313–7317. http://dx.doi.org/10.1016/j.eswa.2008.09.049.

Lv, D., Wang, D., Li, M., & Xiang, Y. (2020). DNN models based on dimensionality reduction for stock trading. *Intelligent Data Analysis*, *24*(1), 19–45. http://dx.doi.org/10.3233/IDA-184403.

Magkonis, G., & Tsouknidis, D. (2017). Dynamic spillover effects across petroleum spot and futures volatilities, trading volume and open interest. *International Review of Financial Analysis*, *52*, 104–118. http://dx.doi.org/10.1016/j.irfa.2017.05.005.

Matuozzo, A., Yoo, P., & Provetti, A. (2023). A right kind of wrong: European equity market forecasting with custom feature engineering and loss functions. *Expert Systems with Applications*, *223*, Article 119854. http://dx.doi.org/10.1016/j.eswa.2023.119854.

Mayhew, S., & Stivers, C. (2003). Stock return dynamics, option volume, and the information content of implied volatility. *Journal of Futures Markets*, *23*(7), 615–646. http://dx.doi.org/10.1002/fut.10084.

Miljkovic, D., & Goetz, C. (2020). The effects of futures markets on oil spot price volatility in regional US markets. *Applied Energy*, (273), http://dx.doi.org/10.1016/j.apenergy.2020.115288.

Niu, H., & Xu, K. (2020). A hybrid model combining variational mode decomposition and an attention-GRU network for stock price index forecasting. *Mathematical Biosciences and Engineering*, *17*(6), 7151–7166. http://dx.doi.org/10.3934/MBE.2020367.

Onan, M., Salih, A., & Yasar, B. (2014). Impact of macroeconomic announcements on implied volatility slope of SPX options and VIX. *Finance Research Letters*, *11*(4), 454–462. http://dx.doi.org/10.1016/j.frl.2014.07.006.

Pan, Z., Wang, Y., Liu, L., & Wang, Q. (2019). Improving volatility prediction and option valuation using VIX information: A volatility spillover GARCH model. *Journal of Futures Markets*, *39*(6), 744–776. http://dx.doi.org/10.1002/fut.22003.

Panagiotidis, T., Stengos, T., & Vravosinos, O. (2018). On the determinants of bitcoin returns: A LASSO approach. *Finance Research Letters*, *27*, 235–240. http://dx.doi.org/10.1016/j.frl.2018.03.016.

Pinto, J., Neves, R., & Horta, N. (2015). Boosting trading strategies performance using VIX indicator together with a dual-objective evolutionary computation optimizer. *Expert Systems with Applications*, *42*(19), 6699–6716. http://dx.doi.org/10.1016/j.eswa.2015.04.056.

Qiu, M., Song, Y., & Akagi, F. (2016). Application of artificial neural network for the prediction of stock market returns: The case of the Japanese stock market. *Chaos, Solitons & Fractals*, *85*, 1–7. http://dx.doi.org/10.1016/j.chaos.2016.01.004.

Rich, D., Livingston, K., & Morgan, S. (2020). Evaluating performance of lasso relative to PCA and LDA to classify dyes on fibers. *Forensic Chemistry*, (18), http://dx.doi.org/10.1016/j.forc.2020.100213.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, *58*(1), 267–288. http://dx.doi.org/10.1111/j.2517-6161.1996.tb02080.x.

Tsai, C., & Hsiao, Y. (2010). Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches. *Decision Support Systems*, *50*(1), 258–269. http://dx.doi.org/10.1016/j.dss.2010.08.028.

Vo, V., Jiawei, L., & Vo, B. (2013). Dimensionality reduction by turning points for stream time series prediction. *Studies in Computational Intelligence*, *457*, 167–176. http://dx.doi.org/10.1007/978-3-642-34300-1_16.

Vo, N., & Ślepaczuk, R. (2022). Applying hybrid ARIMA-SGARCH in algorithmic investment strategies on S & P500 index. *Entropy*, *24*(2), http://dx.doi.org/10.3390/e24020158.

Wang, C., Lin, S., Huang, H., & Wu, P. (2012). Using neural network for forecasting TXO price under different volatility models. *Expert Systems with Applications*, *39*(5), 5025–5032. http://dx.doi.org/10.1016/j.eswa.2011.11.038.

Wu, D., Wang, X., Su, J., Tang, B., & Wu, S. (2020). A labeling method for financial time series prediction based on trends. *Entropy*, *22*(10), 1–25. http://dx.doi.org/10.3390/e22101162.

Xiao, J., Wen, F., Zhao, Y., & Wang, X. (2021). The role of US implied volatility index in forecasting Chinese stock market volatility: Evidence from HAR models. *International Review of Economics and Finance*, *74*, 311–333. http://dx.doi.org/10.1016/j.iref.2021.03.010.

Yu, X., & Huang, Y. (2021). The impact of economic policy uncertainty on stock volatility: Evidence from GARCH–MIDAS approach. *Physica A: Statistical Mechanics and its Applications*, (570), http://dx.doi.org/10.1016/j.physa.2021.125794.

Zhang, Y., Ding, S., & Scheffel, E. (2018). Policy impact on volatility dynamics in commodity futures markets: Evidence from China. *Journal of Futures Markets*, *38*(10), 1227–1245. http://dx.doi.org/10.1002/fut.21905.

Zhang, S., & Fang, W. (2021). Multifractal behaviors of stock indices and their ability to improve forecasting in a volatility clustering period. *Entropy*, *23*(8), http://dx.doi.org/10.3390/e23081018.

Zhang, Y., & Wu, L. (2009). Stock market prediction of S & P 500 via combination of improved BCO approach and BP neural network. *Expert Systems with Applications*, *36*(5), 8849–8854. http://dx.doi.org/10.1016/j.eswa.2008.11.028.

Zhang, J., & Xiang, Y. (2008). The implied volatility smirk. *Quantitative Finance*, *8*(3), 263–284. http://dx.doi.org/10.1080/14697680601173444.

Zhao, J., Zeng, D., Liang, S., Kang, H., & Liu, Q. (2021). Prediction model for stock price trend based on recurrent neural network. *Journal of Ambient Intelligence and Humanized Computing*, *12*(1), 745–753. http://dx.doi.org/10.1007/s12652-020-02057-0.

Zhong, X., & Enke, D. (2017). Forecasting daily stock market return using dimensionality reduction. *Expert Systems with Applications*, *67*, 126–139. http://dx.doi.org/10.1016/j.eswa.2016.09.027.

Zhu, X., Huang, Z., Yang, Y., Tao Shen, H., Xu, C., & Luo, J. (2013). Self-taught dimensionality reduction on the high-dimensional small-sized data. *Pattern Recognition*, *46*(1), 215–229. http://dx.doi.org/10.1016/j.patcog.2012.07.018.