Expert Systems With Applications 103 (2018) 25–37

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models

Ha Young Kim[a,b,*], Chang Hyun Won[a]

[a] Department of Financial Engineering, Ajou University, Worldcupro 206, Yeongtong-gu, Suwon, 16499, Republic of Korea
[b] Department of Data Science, Ajou University, Worldcupro 206, Yeongtong-gu, Suwon, 16499, Republic of Korea

ARTICLE INFO

Article history:
Received 25 November 2017
Revised 10 February 2018
Accepted 1 March 2018
Available online 6 March 2018

Keywords:
LSTM
GARCH
Deep learning
Volatility prediction
Hybrid model

ABSTRACT

Volatility plays crucial roles in financial markets, such as in derivative pricing, portfolio risk management, and hedging strategies. Therefore, accurate prediction of volatility is critical. We propose a new hybrid long short-term memory (LSTM) model to forecast stock price volatility that combines the LSTM model with various generalized autoregressive conditional heteroscedasticity (GARCH)-type models. We use KOSPI 200 index data to discover proposed hybrid models that combine an LSTM with one to three GARCH-type models. In addition, we compare their performance with existing methodologies by analyzing single models, such as the GARCH, exponential GARCH, exponentially weighted moving average, a deep feedforward neural network (DFN), and the LSTM, as well as the hybrid DFN models combining a DFN with one GARCH-type model. Their performance is compared with that of the proposed hybrid LSTM models. We discover that GEW-LSTM, a proposed hybrid model combining the LSTM model with three GARCH-type models, has the lowest prediction errors in terms of mean absolute error (MAE), mean squared error (MSE), heteroscedasticity adjusted MAE (HMAE), and heteroscedasticity adjusted MSE (HMSE). The MAE of GEW-LSTM is 0.0107, which is 37.2% less than that of the E-DFN (0.017), the model combining EGARCH and DFN and the best model among those existing. In addition, the GEW-LSTM has 57.3%, 24.7%, and 48% smaller MSE, HMAE, and HMSE, respectively. The first contribution of this study is its hybrid LSTM model that combines excellent sequential pattern learning with improved prediction performance in stock market volatility. Second, our proposed model markedly enhances prediction performance of the existing literature by combining a neural network model with multiple econometric models rather than only a single econometric model. Finally, the proposed methodology can be extended to various fields as an integrated model combining time-series and neural network models as well as forecasting stock market volatility.

1. Introduction

Volatility in asset markets, such as the stock market, refers to the degree to which asset prices fluctuate and measures the degree of uncertainty or risk. Investment firms and private investors measure risk using the volatility of underlying asset prices (Markowitz, 1952). A derivative good's price can be determined by the volatility of the underlying assets, while portfolio risk can be measured only by the volatility of constituent assets. The accurate prediction of volatility is important to effectively determine the value of derivative goods and hedge against underlying assets. Volatility is important not only in evaluating such intangible derivative goods as stock index options but also in the pricing of equity-linked securities, which are over-the-counter derivatives. Moreover, various value-at-risk models, which measure market risk by calculating the maximum loss that can occur during a target period given a level of confidence in normal market conditions, require some volatility forecasting. Hence, volatility in financial markets plays an important role in derivative pricing, portfolio risk management, and hedging strategies. Studies that predict and model financial markets' volatility are critical, and as a result, volatility has become a topic of extensive research.

Various studies have been conducted to predict the volatility of returns using financial time-series models. The regression model assumes homoskedasticity where the variance of the error term is constant over time. However, since heteroscedasticity is one of the common characteristics of the financial time series, the financial times series model assumes a heteroscedastic condition in which past information affects the future (Hamilton, 1994).

* Corresponding author at: Department of Financial Engineering, Ajou University, Worldcupro 206, Yeongtong-gu, Suwon, 16499, Republic of Korea.
E-mail addresses: hayoungkim@ajou.ac.kr, qqq@ajou.ac.kr (H.Y. Kim).

https://doi.org/10.1016/j.eswa.2018.03.002
0957-4174/

Engle (1982) proposed an autoregressive conditional heteroscedasticity (ARCH) model in which the conditional variance is obtained by expressing the error term at the present time as a function of the previous time. The generalized autoregressive conditional heteroscedasticity (GARCH) model proposed by Bollerslev (1986) generalizes the ARCH model and reduces the estimated number of parameters by expanding it. Financial time-series data exhibit "volatility clustering" behavior (Mandelbrot, 1967), a phenomenon where when volatility is high, this highly volatile state is maintained to some extent, and when volatility is low, this low volatility state is maintained up to a certain threshold. Furthermore, financial time-series data generally have a probability distribution with a fat-tail shape and a peak with a sharper edge than a normal distribution, or leptokurtosis. The ARCH and GARCH models both capture volatility clustering and leptokurtosis. However, such negative shocks are limited in that they cannot capture the asymmetries of volatility such as the leverage effect that exerts a greater influence on volatility than a positive shock of the same magnitude. Nelson (1991) overcame this limitation by proposing the exponential GARCH (EGARCH) model. The GARCH (1, 1) model is particularly easy to handle and superior to other financial time-series models (Bollerslev, 1987). Moreover, Wilhelmsson (2006) previously demonstrated the GARCH (1, 1) model's predictive power through returns from the S&P 500 index futures. Morgan and Reuters (1996) proposed the exponentially weighted moving average (EWMA) model based on the GARCH (1, 1) model. The former emphasizes recent data more than other models do, is appropriate for capturing short-term changes, and is less influenced by the quantity of data, as it uses only a small number of samples from the dataset. However, a disadvantage of the model is its difficulty capturing long-term features.

These econometric methods are advantageous in that they can be theoretically described based on statistical logic, but they contain the assumption that explanatory variables must always be stationary, and when the qualitative variables are mixed, the performance of the models deteriorates remarkably. An artificial neural network (ANN) is a type of machine-learning algorithm and a data-driven nonparametric method. As it has fewer restrictions and assumptions on modeling compared to conventional econometric methods, it can be modeled if enough data exist. This behavior can be explained by the universal approximation theorem (Cybenko, 1989), which states that all finite, continuous functions can be approximated by ANNs. Furthermore, ANNs are effective in modeling nonlinear relationships. Ormoneit and Neuneier (1996) used a multilayer perceptron and density-estimating neural network to predict the volatility of the DAX German index. A comparison of the two models revealed that the density-estimating neural network without a specific target distribution demonstrated better performance than the multilayer perceptron did. Gonzalez Maranda and Burgess (1997) modeled the implied volatility of IBEX 35 index options with a multilayer perception neural network, indicating that it could capture the characteristics of the nonlinear relationships that help predict volatility. Hamid and Iqbal (2004) used an ANN to predict the volatility of S&P 500 index futures and empirically proved that the predictions from ANNs are better than implied volatility estimates. Baruník and Křehlík (2016) proposed the ANN to forecast energy market volatility and they improved volatility forecast by combining the feature of high frequency data with the ANN. Oliveira, Cortez, and Areal (2017) proposed a methodology that predicts returns, volatility, and trading volume by extracting sentiment and attention indicators from microblogging data. They tested five different machine learning algorithms including neural networks and compared them with the Diebold–Mariano (DM) test. Thus, neural network-based methodologies can clearly extract nonlinear features that cannot be captured in the econometric model in volatility predictions and these have demonstrated outstanding performance, particularly in the financial time-series model.

Unlike feedforward neural networks, recurrent neural networks (RNNs) have feedback connections inside the neural network to learn temporal patterns. Therefore, RNNs can be used for effective sequential data modeling and time-series analyses. Maknickiene and Maknickas (2012) improved prediction performance over feedforward neural networks using a long short-term memory (LSTM) model, a special type of RNN, to predict exchange rates and foreign exchange trading, which are examples of financial market time series. Chen, Zhou, and Dai (2015) used an LSTM model to predict returns in the Chinese stock market, confirming that the LSTM model demonstrated better performance than the random prediction method did. Thus, it has been observed in the literature that the LSTM is superior to the feedforward neural network model as a financial time-series model.

Models combining a neural network model with an econometric model have been studied as well as models predicting volatility with a single neural network model. Roh (2007) presented an integrated of a financial time-series model and a feedforward neural network model using the KOSPI 200 index. A model that combines one feedforward neural network and one financial time-series model, such as the EWMA, GARCH, and EGARCH, has been proven to be superior to the GARCH single model. Moreover, the model combining the feedforward neural network with the EGARCH model had the best results. Tseng, Cheng, Wang, and Peng (2008) integrated an EGARCH model and a feedforward neural network to estimate the volatility of Taiwan stock index option prices. Their results reduced the stochasticity and nonlinearity in the error term and showed better performance in the integrated model than the EGARCH model. Hajizadeh, Seifi, Zarandi, and Turksen (2012) presented two hybrid models combining EGARCH and a feedforward network; the result of this integrated econometric model indicated a better result—a smaller error—compared to a single econometric model and a single ANN model. Hajizadeh et al. (2012) presented two kinds of hybrid models integrating an EGARCH model and a feedforward network model. The hybrid model showed better results (smaller prediction error) than a single econometric model and a single ANN model. Kristjanpoller, Fadic, and Minutolo (2014) studied a hybrid neural networks-GARCH model for forecasting volatility in three Latin American markets and showed that the hybrid neural network model reduced the mean absolute percentage error (MAPE) compared to the GARCH model. Kristjanpoller and Minutolo (2016) used a hybrid of ANN and GARCH models to predict volatility in oil prices. They not only added input variables such as the index, the exchange rate associated with oil prices, but also found the best architecture to fit different time windows of volatility. This study demonstrated that the ANN-GARCH model improved the prediction of the GARCH model by 30.6%. Hernández (2017) compared the predictability of GARCH-type models and hybrid neural network models and showed that hybrid neural networks were more suitable than the GARCH models for forecasting volatility of main metals.

The standard models widely used for estimating the volatility of the financial sector, especially the GARCH-type models, reflect the empirical characteristics (volatility clustering, heavy-tailed distribution, etc.) of the in-sample volatility, but these empirical properties for out-of-sample are hard to reproduce (Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. 2003; Corsi, 2009). To improve this, Andersen and Bollerslev (1998) first proposed the realized volatility, which does not depend on the specific assumptions taken by the model used to measure the volatility and reduces measurement errors by using high fre-

quency data. It was confirmed that the accuracy of the estimate improved by reducing the influence of the noise according to the market microstructure. In this regard, (Andersen, T. G., Bollerslev, T., Diebold, F. X., & Ebens, H. 2001a; Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. 2001b) showed that realized volatility is useful in forecasting volatility in the foreign exchange market and the stock market. As the research on realized volatility has become more active, McAleer and Medeiros (2008) summarized the issues related to realized volatility, especially emphasizing models that predicted the daily realized volatility. Hillebrand and Medeiros (2010) used the log linear specification in the heterogeneous autoregressive model and the nonparametric interpretation of the neural network model to predict the realized volatility. The experimental results show that the log linear model predicts better than the nonlinear model. McAleer and Medeiros (2011) compared the nonlinear heterogeneous autoregressive model based on the neural network (nonlinear HAR-NN) with the linear heterogeneous autoregressive model (HAR) to predict the daily realized volatility of the S&P 500 and FTSE 100 futures. As a result, the nonlinear HAR-NN outperformed the benchmark HAR model when evaluated as mean squared error. Baruník and Křehlík (2016) proposed a method of finding hidden features in predicting the realized volatility of an energy market with a generalized regression based on a neural network. Experimental results showed robust performance in predicting realized volatility despite a data period such as the 2008 financial crisis. Yao et al. (2017) constructed an autoregressive neural network and autoregressive model based on the neural network using decomposed volatility data and proposed a hybrid model that aggregated output from both models. They showed that the hybrid model had better performance in the realized volatility prediction than the single model GARCH, EGARCH, and neural network model.

Financial time-series models, such as GARCH, EGARCH, and EWMA, have both advantages and disadvantages in forecasting stock market volatility, and they use various characteristics, such as leverage effects, excess kurtosis, and volatility clustering, from the financial time-series data. Thus, instead of combining a single econometric model with a single neural network model as in previous studies, this study proposes combining the information obtained from various financial time-series models with a neural network model. In other words, this study proposes a hybrid LSTM model with multiple GARCH type models, which is a volatility prediction model with parameters of two or more GARCH-type models as inputs to the LSTM model. Our conjecture is that this would be more effective in predicting financial market volatility. Therefore, rather than combining an econometric model and a neural network model as in previous studies, combining various information from several econometric models with a neural network model would be more effective in predicting financial market volatility.

We predict the realized volatility of the KOSPI (Korea Composite Stock Price Index) 200 index using the proposed model. The KOSPI 200 index is composed of 200 stocks with high market cap and large trading volume. The KOSPI 200 accounts for about 20% of all listed companies but the total market capitalization accounts for more than 70%. As the KOSPI 200 index options market opened on July 7, 1997, it has become one of the biggest single markets in terms of trading volume, and the KOSPI 200 index options market is one of the most active derivatives markets worldwide by volume. Therefore, it is critical to predict the KOSPI 200 index volatility in Korea accurately to manage risk in the Korean financial market. As a result, many studies have been conducted on the KOSPI 200 index (Kim, 2006; Roh, 2007; Park, Lee, Song, & Chun, 2010; Ryu, 2012; Park. Kim, & Lee, 2014; Hsu, Lessmann, Sung, Ma, & Johnson, 2016; Shim, Kim, & Ryu, 2017). The proposed hybrid models are generated in this study using EWMA, GARCH, and EGARCH financial time-series volatility models, as these are widely used to predict the volatility of the KOSPI 200 index.

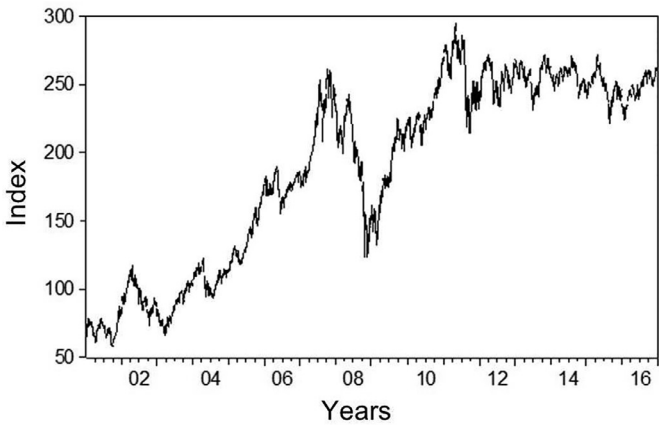

Fig. 1. KOSPI 200 index.

The remainder of this paper is organized as follows. Section 2 outlines the data used for the model and explains GARCH-type financial time-series models as well as ANNs and the proposed hybrid LSTM model with multiple GARCH-type models. Section 3 contains information on how the experiments are conducted. Section 4 reveals the experimental results and discussion. Section 5 presents conclusions and suggests future research.

2. Materials and methods

2.1. Data

This study considers the volatility of KOSPI 200 stock index returns, and tracks the market capitalization of 200 stocks representing Korea as a fundamental analysis target. The data, obtained from the data guide, consist of 2,665 data points from January 1, 2001, to September 30, 2011; and 1,298 data points are predicted until January 2, 2017. The 3-year Korea Treasury Bond (KTB) interest rate and 3-year AA-grade corporate bond (CB) interest rate are based on daily data provided by Korea Asset Management Corporation. The daily closing prices of gold and crude oil were from Bloomberg for the same period. Table 1 shows common descriptive statistics such as mean, standard deviation, skewness, and kurtosis of the time series data used in this study and also shows the results of the augmented Dickey–Fuller (ADF) test, which is a stationary test, and the Jarque-Bera test, which is a normality test.

Fig. 1 illustrates the upward tendency in the KOSPI 200 index. An ADF test was used to confirm the stability of this time series. The null hypothesis (h_0) of the ADF test is that the corresponding time series has a unit root. As shown in Table 1, the ADF test statistic is −1.63, but the 10% critical value is −2.56, so the null hypothesis cannot be rejected and, hence, the time series has a unit root, which is unstable (Dickey and Fuller, 1979). A nonstationary time series should be transformed into a stationary time series by variable conversion or differentiation to be used for financial time-series models.

We convert this to a stationary time series by using a return value that takes the difference of the logarithmic value of the original time series. In Table 1, the ADF test statistics for this logarithmic return show −62.2, which is less than the 10% critical value of −2.56, so h_0 can be rejected. As shown in Fig. 2, the logarithmic return time series oscillates based on the average value of 0 and is stationary.

As Fig. 3 and Table 1 demonstrate, the distribution of KOSPI index returns reveals high kurtosis and negative skewness, similar

Table 1
Summary statistics for explanatory variables: *** and ** denote a rejection of the null hypothesis at the 1% and 5% significance level, respectively.

	Mean	Standard deviation	Skewness	Kurtosis	Jarque-Bera	ADF
KOSPI 200 INDEX	189.47	68.18	−0.47	1.73	416.02***	−1.63
Gold price	906.65	463.03	0.14	1.72	282.50***	−1.30
Oil price	64.65	27.79	0.21	2.02	186.29***	−1.85
CB interest rate	4.56	1.55	−0.02	2.42	55.45***	−1.61
KTB interest rate	3.92	1.32	−0.20	2.27	115.74***	−1.74
KOSPI 200 INDEX log difference	0.00015	0.0065	−0.39	8.81	5675.93***	−62.23***
	ARCH-LM Test (Lag = 5)		F-statistic	2.236**		
			Obs*R-squared	11.17**		

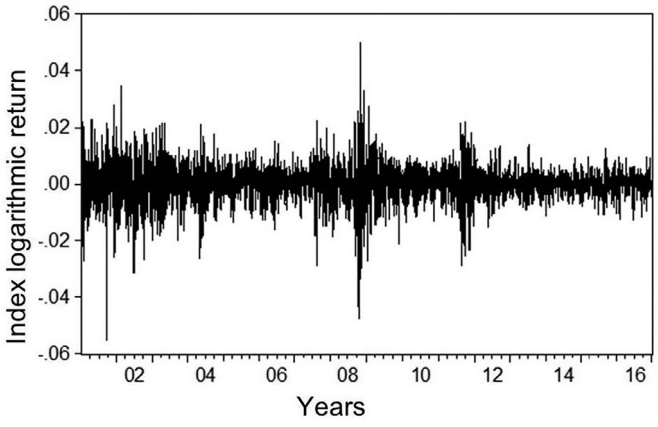

Fig. 2. KOSPI 200 index logarithmic return.

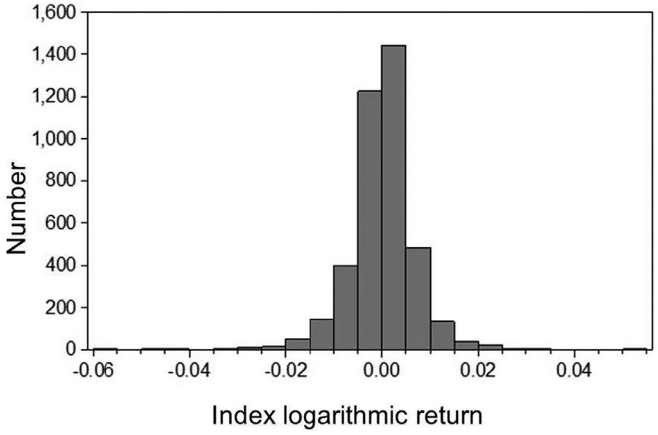

Fig. 3. Log difference in KOSPI 200 index histogram.

to other market variables. In other words, the distribution has a thicker tail than the normal distribution. In addition, the Jarque–Bera test rejects the normality of the return distribution as the result has a value of 5675.93 under the condition that the thresholds are 5.99 and 9.21 at 5% and 1% significance levels, respectively. Therefore, it can be assumed that dependency or heterogeneity exists in the time series, as the distribution differs from the normal distribution. As Table 1 indicates, the presence of heteroscedasticity is verified by the ARCH-Lagrange multiplier (ARCH-LM) test (Bollerslev, 1986).

The hypothesis that there is no conditional heteroscedasticity until the fifth lag of the null hypothesis of the ARCH-LM test is rejected with a significance level of 5%. Hence, heteroscedasticity exists until the fifth lag. Therefore, we determine that the volatility prediction model of the time series, which assumes such heteroscedasticity as in the ARCH and GARCH models, is a suitable model for KOSPI index returns.

2.2. Financial time-series models

2.2.1. GARCH (1, 1) model

It is assumed in the regression analysis that variance is constant. Under the assumption that the probability distributions of the residual terms are identical in all observations, if the conditional variance is constant, the estimator is unbiased but cannot be used for valid estimation. Consequently, the reliability of commonly used tests and the settings of the confidence intervals cannot be obtained. Engle (1982) proposed a p-order autoregressive conditional heteroscedasticity model, or ARCH (p), to model time series with volatility clustering or fat-tail characteristics from the perspective of conditional distribution, as demonstrated in Eqs. (1)–(3).

$$y_t = \mu_t + \sigma_t \eta_t,\ \eta_t \sim \mathrm{N}(0,1) \tag{1}$$

$$\varepsilon_t = \sigma_t \eta_t,\ \varepsilon_t | \chi_{t-1} \sim \mathrm{N}\left(0, \sigma_t^2\right) \tag{2}$$

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \cdots\cdots + \alpha_q \varepsilon_{t-q}^2 \tag{3}$$

Eq. (1) comprises μ_t, which can be predicted by the average equation for the y_t time series, and the unpredictable error term ε_t. Eq. (2) demonstrates that the error term ε_t follows normal distribution under the condition that all information at time $(t-1)$ is known. In addition, we observe in Eq. (3) that conditional variance σ_t^2 at time t depends on the past squared residual values; every estimated parameter must be nonnegative for the conditional variance to be positive.

The disadvantage of the ARCH (p) model is that the method for determining the order (p) is problematic and a number of parameters are needed if the p-value is too large. Bollerslev (1986) proposed a generalized model to solve these problems, called the GARCH model. The GARCH (p, q) model is as follows:

$$y_t = \mu_t + \sigma_t \eta_t,\ \eta_t \sim \mathrm{N}(0,1) \tag{4}$$

$$\varepsilon_t = \sigma_t \eta_t,\ \varepsilon_t | \chi_{t-1} \sim \mathrm{N}\left(0, \sigma_t^2\right) \tag{5}$$

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \cdots\cdots + \alpha_q \varepsilon_{t-q}^2 + \beta_1 \sigma_{t-1}^2 + \cdots\cdots + \beta_p \sigma_{t-p}^2 \tag{6}$$

This model generalizes the ARCH model so that the conditional variance, σ_t^2, includes not only the square terms of the past error term but also the conditional variance terms, as shown in Eq. (6). Thus, the volatility is predicted by taking the weighted sum of the variance predicted from the past and volatilities observed from the past.

The GARCH (1, 1) model can be expressed as in Eq. (7):

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \quad (7)$$

By rearranging the equation, we obtain Eq. (8):

$$\sigma_t^2 = \frac{\alpha_0}{(1-\beta_1)} + \alpha_1 \sum_{j=1}^{\infty} \beta_1^{j-1} \varepsilon_{t-j}^2 \quad (8)$$

This model predicts the volatility of the current period for KOSPI 200 index returns by using a weighted sum of the predicted variance and the observed volatility from the past. We give geometrically decreasing weights to the square of the error terms that are distances away, which is consistent with volatility clustering, a characteristic of the financial time series.

2.2.2. EGARCH model

The EGARCH model proposed by Nelson (1991) does not require every coefficient of the dispersion equation to be nonnegative. Moreover, this model can incorporate the leverage effect, which reflects the asymmetric impacts of negative and positive impacts of the same magnitude. The EGARCH model is defined as follows:

$$r_t = X_t \mathrm{M} + \varepsilon_t \quad (9)$$

$$\ln \sigma_t^2 = \alpha_0' + \beta \ln \sigma_{t-1}^2 + \omega\left(\frac{\varepsilon_{t-1}}{\sigma_{t-1}}\right) + \gamma \left|\frac{\varepsilon_{t-1}}{\sigma_{t-1}}\right| \quad (10)$$

The time series r_t in Eq. (9) is composed of explanatory variable X, parameter M, and error term ε_t. This model guarantees that the conditional variance is positive even if the estimated parameter is negative. The parameter for measuring the leverage effect is ω. Consequently, if $\frac{\varepsilon_{t-1}}{\sigma_{t-1}} < 0$, then $\gamma - \omega$ is obtained, and if $\frac{\varepsilon_{t-1}}{\sigma_{t-1}} > 0$, then this yields $\gamma + \omega$, which is reflected asymmetrically.

2.2.3. EWMA model

RiskMetrics, by J.P. Morgan & Co., uses an EWMA model that exponentially reduces the weights of information from the past and calculates volatility using a rolling window technique. The EWMA is defined as follows:

$$\sigma_t^2 = \rho \sigma_{t-1}^2 + (1-\rho)\varepsilon_{t-1}^2 \ (0 < \rho < 1) \quad (11)$$

The model denoted by Eq. (11) predicts conditional variance σ_t^2, where ρ is the degree of weighting decrease, and gives heavier weights to recently acquired information. This makes the model more responsive to changes by providing greater weights for the most recent data and exponentially decreasing weights for past data. However, an excessively large ρ is disadvantageous in that it reflects only the most recent information and ignores long-term memories.

2.3. Neural network models

2.3.1. Deep feedforward network

An ANN is one of the machine-learning algorithms designed to mimic the structure of neurons in the brain. These networks have networking capabilities in which artificial neurons are connected to each other and they acquire a problem-solving ability by adjusting their connection strengths through learning. A deep feedforward network (DFN) is a representative deep learning model. This neural network model sequentially calculates the output layer from the input layer using the output of the previous layer as the input of the current layer (Fig. 4).

$$\hat{v}_t = f\left(f\left(f\left(x_i w_{ji} + b_j\right) w_{mj} + b_m\right) w_{km} + b_k\right) \quad (12)$$

Eq. (12) demonstrates that input value x_i is multiplied by weight w_{ji} and summed with bias b_j; the result is added to the next layer. $f(\cdot)$ is an activation function, or typically, a differentiable nonlinear function, such as a sigmoid function. As Eq. (12) indicates, the weight matrix is repeatedly multiplied and summed with the bias term, and the activation function is applied between every layer until the output layer is reached. A DFN uses a supervised learning method and the weights are learned by minimizing the difference between the predicted value $\hat{v}_t$ and the target value v_t of the DFN. The DFN-based model in this study used four different loss functions defined in Section 2.5.2. A backpropagation algorithm is used to reduce the error of the value calculated by the forward propagation and optimizes the objective function by using a gradient descent to update the weights.

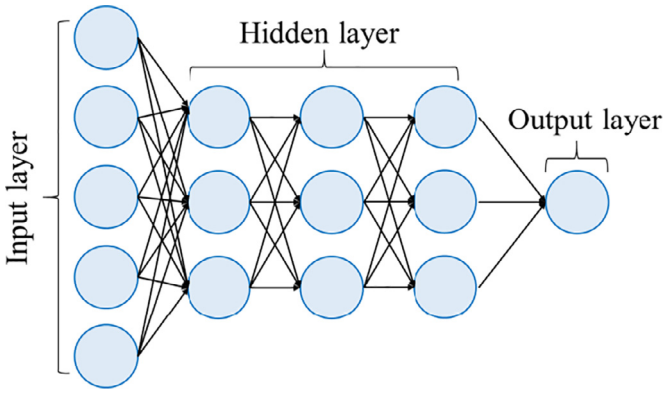

Fig. 4. A deep feedforward network.

2.3.2. Long short-term memory

A recurrent neural network is one that learns sequential patterns through internal loops by receiving input sequences. Backpropagation is performed to learn the weights, and the slope calculated by the chain rule must be propagated. As the values are backpropagated into the activation function, such as the sigmoid and tanh functions, the slope becomes extremely small (or extremely large) and encounters the problem of vanishing (or exploding gradients). Backpropagation is vulnerable to long-range dependency. LSTM models were developed to avoid these problems. Hochreiter and Schmidhuber (1997) introduced LSTMs that used memory cells and gates to store information for long periods of time, or to forget unnecessary information.

$$g_t = \sigma\left(U_g x_t + W_g h_{t-1} + b_f\right) \quad (13)$$

$$i_t = \sigma\left(U_i x_t + W_i h_{t-1} + b_i\right) \quad (14)$$

$$\tilde{c}_t = \tanh\left(U_c x_t + W_c h_{t-1} + b_c\right) \quad (15)$$

$$c_t = g_t * c_{t-1} + i_t * \tilde{c}_t \quad (16)$$

$$o_t = \sigma\left(U_o x_t + W_o h_{t-1} + b_o\right) \quad (17)$$

$$h_t = o_t * \tanh(c_t) \quad (18)$$

LSTMs are composed of memory blocks instead of neurons. As Fig. 5 illustrates, the LSTM is composed of a memory cell (c_t) and three gates: an input gate (i_t), a forget gate (g_t), and an output gate (o_t). At time t, x_t represents the input and h_t the hidden state. Symbol $\otimes$ denotes the point-wise multiplication. $\tilde{c}_t$, also called an input modulate gate, is a value that determines how much new information is received in the cell state. The three gates, the cell state (c_t), and the hidden state (h_t) are calculated as shown in Eqs. (13)–(18). In these equations, U and W are weight matrixes, b is a bias term,

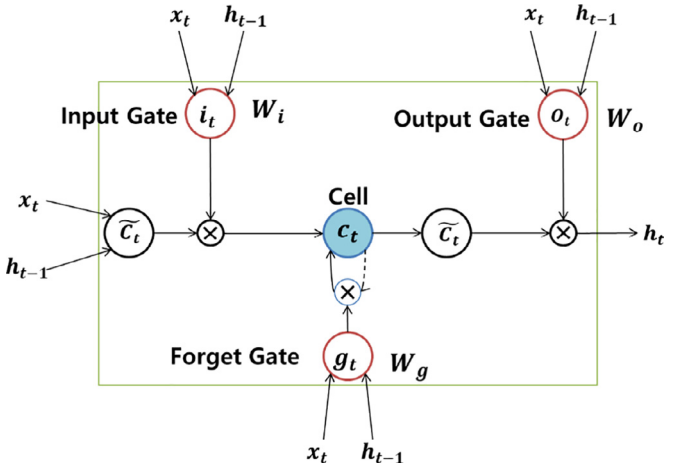

Fig. 5. An LSTM memory block.

$\sigma(\cdot)$ is a sigmoid function, and symbol $*$ denotes element-wise multiplication.

First, the forget gate g_t of Eq. (13) produces the weighted sum of x_t, h_{t-1}, and bias as a value from 0 to 1 through the sigmoid function. A value of one passed through the gate means that all input information passes through the gate, and a value of zero means that no input information is passed. Thus, the forget gate controls the amount of information in the past cell state (c_{t-1}) in updating the cell state at time t, as shown in Eq. (16). Eq. (14) is the calculation formula of input gate i_t and determines how much new information is stored in the cell state (c_t). Eq. (15) calculates the new information at time t and its output through the tanh function has a value between -1 and 1. Past cell state information and new information, which are controlled by the forget and input gates, are calculated as variable c_t of time t, as noted in Eq. (16). Finally, output value h_t is determined by passing through the output gate o_t (Eq. (15)) and filtering at c_t, while c_t passes into the tanh function so that the value is between -1 and 1. The selected values are converted to output by multiplying them by o_t. This process updates the cell state c_{t-1}, necessary information is separated from unnecessary information, and output becomes h_t as noted in Eq. (18). The LSTM model consisting of these memory blocks is learned by using backpropagation through the time algorithm.

2.4. Proposed hybrid models – LSTM with multiple GARCH models

Roh (2007) fused a shallow feedforward neural network and a financial time-series model to improve the ANN learning process and its volatility predictions. As explained in the introduction, such financial time-series models as the GARCH, EGARCH, and EWMA have been widely used to predict financial market volatility. Previous studies by Roh (2007), Tseng et al. (2008), Hajizadeh et al. (2012), Kristjanpoller et al. (2014), Kristjanpoller and Minutolo (2016), and Hernández (2017) integrated feedforward neural networks and financial time-series models to strengthen volatility predictions rather than a single GARCH-type model.

As mentioned in the introduction, the EWMA model is suitable for capturing short-term changes and the GARCH model for capturing volatility clustering and leptokurtosis information, while the EGARCH model is useful for leverage effect modeling. Hence, each GARCH-type model has advantages and disadvantages in its volatility prediction. Therefore, combining two or more GARCH-type models to reflect various characteristics, rather than combining a single GARCH-type model with a feedforward neural network, would enhance volatility predictions.

In this study, to verify this hypothesis, we propose a new hybrid model by adding parameters of two or more GARCH-type models as an input of neural networks. Compared to the traditional econometric methodologies, the neural network models use fewer assumptions, have fewer modeling constraints, and learn high-level features on their own. These characteristics help the neural network to make predictions from the input data. The following clarifies in more detail how information increases as parameters are added. In Eq. (7) of the GARCH (1, 1) model, α_1 indicates the magnitude of the volatility shock, that is, the magnitude of the effect of the current volatility shock on the volatility at the next time point, and β_1 indicates the persistence of the past volatility. In Eq. (10), the EGARCH model is determined by the sign and magnitude of $\frac{\varepsilon_{t-1}}{\sigma_{t-1}}$, and if $\frac{\varepsilon_{t-1}}{\sigma_{t-1}}$ is positive, then the $\omega+\gamma$ effect is obtained; if it is negative, the $\omega-\gamma$ effect is obtained; thereby, the model reflects that the volatility asymmetrically responds to positive shocks and negative shocks. In other words, not only the magnitude of the volatility shock but also the direction of the volatility shock can be known by using ω (leverage) and γ (leverage effect). In addition, since the magnitude of the volatility shock obtained from the EGARCH is also calculated by reflecting the leverage effect, it will be different from the magnitude shock information obtained from GARCH (1, 1). In Eq. (11), the EWMA model is similar to the GARCH (1, 1) model but uses ρ, which has a different meaning from β_1, that is, it gives more weight to the near past. Therefore, if all the parameters estimated from GARCH (1, 1), EGARCH and EWMA are used in the hybrid model, such as the magnitude of the volatility shock, the persistence of the volatility, the direction of the volatility shock, the magnitude of the volatility shock reflecting the leverage effect, the persistence of the volatility reflecting the leverage effect, the magnitude of the volatility shock reflecting the short-term movements, and the persistence of the volatility reflecting the short-term movements, the prediction performance will be improved as compared to the hybrid model created by only a single econometric model.

Consequently, when we add the parameters obtained from various GARCH-type models as an input, the neural network learns the features that help the prediction through the input variables, including the various pieces of information related to volatility and, thus, the prediction performance of the proposed hybrid model improves. In addition, past studies used a single hidden layer feedforward network model, that is, a shallow feedforward neural network. However, to improve the volatility predictions in this study, we add hidden layers to deepen the model. We use a DFN model, which improves nonlinear prediction power and an LSTM, which learns temporal patterns brilliantly from the time-series data.

Fig. 6 shows an example of an integrated model architecture that combines LSTM with the GARCH model and the EGARCH model among the proposed hybrid models that combine LSTM with various GARCH-type models. Specifically, we use the GARCH, EGARCH, and EWMA models as the GARCH-type models. First, we create three hybrid models that integrate two of the three GARCH-type models with an LSTM network. The model combining GARCH, EGARCH, and LSTM is referred to as GE-LSTM; the model combining GARCH, EWMA, and LSTM is referred to as GW-LSTM; and the model combining EGARCH, EWMA, and LSTM is referred to as EW-LSTM.

These three models are used to predict the volatility of the next day based on 22 trading days. As Fig. 6 illustrates, the GARCH parameter, EGARCH parameters, and explanatory variables from each 22-day time period are used as the LSTM input. As explanatory variables, we use the KOSPI 200 index, the KOSPI 200 index log difference return rate, the time of the KOSPI 200 index log difference return rate at -1 time, the 3-year KTB interest rate, the interest rate for the 3-year AA-grade CB, the price of crude oil, and the price of gold. The proposed hybrid model receives the

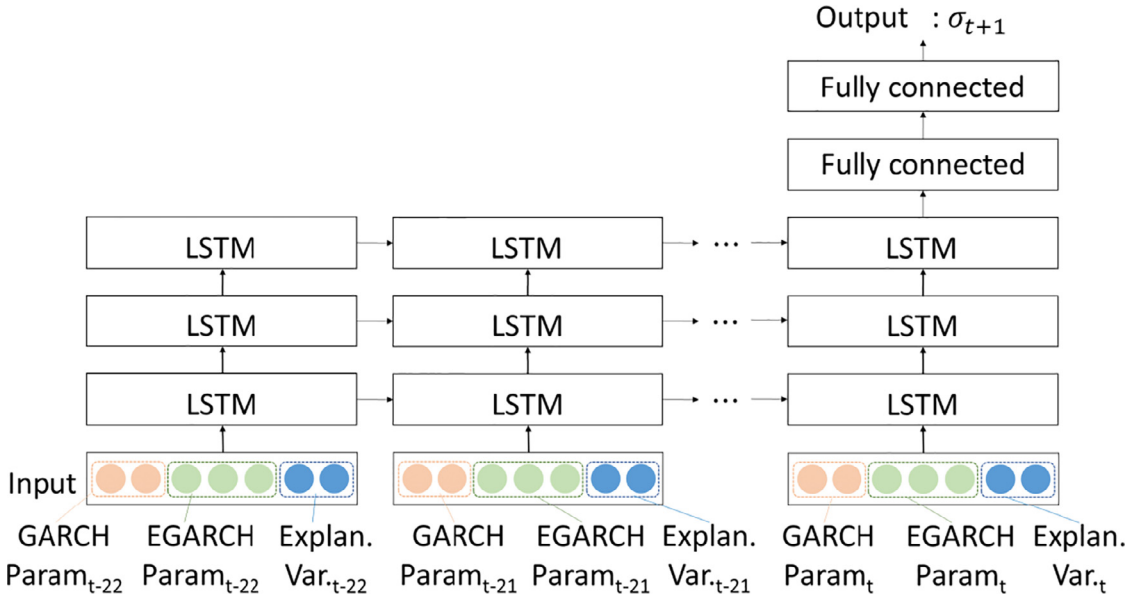

Fig. 6. Architecture of the proposed GE-LSTM hybrid model, created by integrating the GARCH, EGARCH, and LSTM models. *Note:* "Param" refers to parameters, and "Explan. Var." to explanatory variables.

abovementioned variables from time $(t-22)$ to time t, inputs them through three LSTM layers, passes them through two fully connected layers at time t, and then predicts volatility at time $(t+1)$. Moreover, to investigate the effect of combining more GARCH-type models, we create a model that combines all three models, GARCH, EGARCH, and EWMA, with LSTM, and we refer to this model as GEW-LSTM.

2.5. Measurement and statistical test

2.5.1. Realized volatility

We need to compare predicted versus actual volatility, which is set as the target value for the supervised learning process. Realized volatility is the value obtained by observing how much the stock price has changed during a certain period. Realized volatility (RV_t) at time t is calculated using the following equation:

$$RV_t = \sqrt{\frac{1}{\rho_t}\sum_{t=1}^{\rho_t}(s_t - \overline{s_t})^2}, \tag{19}$$

where ρ_t is the number of days remaining after time t, s_t is the log return rate of the KOSPI 200 index at time t, and $\overline{s_t}$ is the average of the log return rates of the KOSPI 200 index, realized during period ρ_t after time t. This standard deviation is used as the actual volatility.

2.5.2. Loss functions

We use different loss functions. Two of the most common metrics, the mean absolute error (MAE) and the mean squared error (MSE), are utilized. In addition, heteroscedasticity adjusted MAE (HMAE) and heteroscedasticity adjusted MSE (HMSE), which are non-linear loss measurements, are used (Fuertes et al., 2009; Kristjanpoller and Minutolo, 2016). These four loss functions are as follows:

$$\text{MAE} = \frac{1}{T}\sum\left|\hat{v}_t - RV_t\right|, \tag{20}$$

$$\text{MSE} = \frac{1}{T}\sum\left(\hat{v}_t - RV_t\right)^2, \tag{21}$$

$$\text{HMAE} = \frac{1}{T}\sum\left|1 - \hat{v}_t/RV_t\right|, \tag{22}$$

$$\text{HMSE} = \frac{1}{T}\sum\left(1 - \hat{v}_t/RV_t\right)^2, \tag{23}$$

where $\hat{v}_t$ is the predicted volatility at time t, RV_t denotes realized volatility at time t, and T is the total number of predictions.

2.5.3. Test for comparing forecast accuracy

We performed the Diebold–Mariano (DM) test and Wilcoxon signed rank (WS) test to verify the equivalence of forecast accuracy for the two forecasting models, that is, to show statistically significant differences in out-of-sample forecast accuracy (Francis, Roberto, 1995). First, let us explain the DM test. We denote forecast errors as $u_{i,t}$, defined by $u_{i,t} = \hat{y}_{i,t} - y_t$, for $i = 1, 2$ where y_t is the actual time-series and $\hat{y}_{i,t}$ is the forecast of the time-series y_t. The null hypothesis of equal accuracy for the two forecasting models is $\mathbb{E}(d_t) = 0$, where $d_t \equiv g(u_{1,t}) - g(u_{2,t})$ is a loss differential with any given loss function $g(\cdot)$. The DM statistics are obtained as follows:

$$DM = \frac{\bar{d}}{\sqrt{2\pi\hat{f}_d(0)/T}}, \tag{24}$$

where $\bar{d} = \frac{1}{T}\sum_{t=1}^{T}(g(u_{1,t}) - g(u_{2,t}))$ and $\hat{f}_d(0)$ is a consistent estimate of $f_d(0)$, which stands for the spectral density of the loss differential at frequency 0. Second, let us explain the WS test. If exact finite-sample tests are available, the observed loss differentials could be tested based on the null hypothesis of the zero-median loss differential: median$(d_t) = 0$. If the sampled loss differential has a symmetrical distribution, then the null hypothesis is in line with the previous DM test. The WS statistics are obtained as follows.

$$\text{WS} = \sum_{t=1}^{T} I_+(d_t)\text{rank}(|d_t|), \tag{25}$$

where $I_+(d_t) = \begin{cases} 1, & d_t > 0 \\ 0, & otherwise \end{cases}$.

3. Experiment

Our experimental procedure was constructed as follows. First, we tested the single models; in other words, we created single

Table 2
Estimated parameters of GARCH variant models.

Models	Trend	ε_{t-1}^2	σ_{t-1}^2	$(\frac{\varepsilon_{t-1}}{\sigma_{t-1}})$	$\lvert\frac{\varepsilon_{t-1}}{\sigma_{t-1}}\rvert$
GARCH	0.000000251	0.067415	0.926795		
EGARCH	− 0.234291		0.987650	− 0.062557	0.138948
EWMA		0.03	0.97		

GARCH-type models (GARCH, EGARCH, and EWMA) and generated the single neural network models (DFN and LSTM) to compare the predictability of volatility between single models. Second, we generated a single GARCH hybrid model, which combined a neural network and single GARCH-type model. This aimed to evaluate the superiority of the LSTM-based integrated model with temporal pattern advantages over the existing integrated model. We then studied the LSTM-based and DFN-based single GARCH hybrid models to compare their performance of volatility predictions. We then constructed double GARCH hybrid models (GE-LSTM, GW-LSTM, and EW-LSTM), which were the two GARCH-type models combined with the LSTM, and we compared their forecast performance with single GARCH hybrid models. Finally, all three GARCH-type models were combined and integrated into an LSTM (GEW-GARCH); we compared its performance with the single and double GARCH hybrid models. The out-of-sample forecast performance was calculated using four different measures, such as MAE, MSE, HMAE, and HMSE, as specified in Section 2.5. In addition, the DW test and the WS test were performed for equal forecast accuracy. The rest of this section provides details of this process.

First, the GARCH, EGARCH, and EWMA parameters were estimated from 2,665 KOSPI 200 index data points, collected between January 1, 2001, and September 30, 2011; Table 2 illustrates the results. The EWMA, in this case, used a common smoothing factor of $\rho = 0.97$ (Morgan, 1996). The models' performance was evaluated by performing an out-of-sample forecasting on each model with 1,298 data points from the KOSPI 200 index from October 4, 2011, to January 2, 2017.

Next, we experimented with the single neural network models: DFN and LSTM. The DFN model's architecture has three hidden layers, and each hidden layer has 10, 5, and 1 neurons, in order from the layer nearest to the input layer. We used an Adam optimizer (Kingma and Ba, 2014) to train the DFN, with the learning rate and epoch set to 0.0001 and 150, respectively. We used five hidden layers for the LSTM model; three were LSTM layers, and two were fully connected. The dropout effect was added to the three LSTM layers to prevent overfitting, and the layers were set to 0.3, 0.8, and 0.8, respectively. In addition, the LSTM layer consisted of 10, 4, and 2 layer nodes, and the fully connected layer consisted of 5 and 1 layer node(s). As with the DFN, we used the Adam optimizer, and set the learning rate and epoch to 0.0001 and 150, respectively. In both neural network models, 20% of the training set was used for the validation set, and the realized volatility was set as the target value.

As mentioned in Section 2.4, we then conducted experiments to compare the single hybrid models based on DFN and LSTM. We used the parameters extracted from the GARCH-type models (Table 2) as inputs, and trained models combining DFN with one of GARCH, EGARCH, and EWMA. We called these hybrid models G-DFN (GARCH-DFN), E-DFN (EGARCH-DFN), and W-DFN (EWMA-DFN). We discovered the number of nodes in the DFN model's hidden layer using trial and error, then narrowed the number of nodes down to 10, 5, and 1 to discover those with the highest accuracy based on the input dimensions. As with the DFN-based single GARCH hybrid model, the LSTM-based hybrid models were also experimented using the G-LSTM (GARCH-LSTM), E-LSTM (EGARCH-LSTM), and W-LSTM (EWMA-LSTM) models. We used 10, 4, 2, 5, and 1 nodes for the LSTM layer and established a model to learn in the direction with the highest accuracy based on the given input dimensions.

In addition, the proposed models in this study were tested using two or more GARCH variant models combined with the LSTM model, such as the GE-LSTM, GW-LSTM, EW-LSTM, and GEW-LSTM. The architecture was identical to the aforementioned single models, but the number of nodes was found through trial and error to discover those with the highest accuracy as the input dimensions differed.

Table 3 summarizes each model's input variables to further examine the models used in this study. This table also outlines the explanatory variables common to all models, such as the KOSPI 200 index, the KOSPI 200 index log difference return rate, the KOSPI 200 index differential rate of return at −1 time, the 3-year KTB interest rate, the interest rates for the 3-year AA-grade corporate bonds, the price of crude oil, and the price of gold. The parameters estimated from the GARCH model ($0.926795\sigma_{t-1}^2$ and $0.067415\varepsilon_{t-1}^2$) were referred to as "GARCH param.," with the numbers in parentheses indicating the number of parameters. The parameters extracted from EGARCH $0.987650\sigma_{t-1}^{2'}$, the leverage effect ($-0.062557(\varepsilon_{t-1}/\sigma_{t-1})$), and the asymmetric leverage effect ($0.138948|\varepsilon_{t-1}/\sigma_{t-1}|$) were expressed as "EGARCH param.," and the EWMA parameter was referred to as the "EWMA param." Symbol O denotes input variables in the model, and X denotes that the input variable was not used.

4. Results and discussion

We created models for the one-day ahead prediction by using four different loss functions of MAE, MSE, HMAE, and HMSE and using the 22-day window and explanatory variables as input data. The out-of-sample forecast errors of the single GARCH-type models, the DFN-based hybrid models, and the LSTM-based hybrid models are summarized in Table 4. Based on these results, the p-values of DM and WS equal predictive accuracy tests for the DFN-based hybrid models and the LSTM-based hybrid models are shown in Table 5. First, when comparing the results of GARCH type models, as in the results of previous studies, the EGARCH model was the smallest in all error measures. The results from the EGARCH model were 0.46% (MAE), 7.5% (MSE), 29% (HMAE), and 45.7% (HMSE); better than in the GARCH model of 10.1% (MAE), 31.5% (MSE), 39.1% (HMAE), and 59.8% (HMSE) and better than in the EWMA. However, the DFN's results were 64%, 11.1%, 40.7%, and 49.6%, for MAE, MSE, HMAE, and HMSE, respectively, and therefore, better than the EGARCH model. Moreover, the results from the LSTM were 74.6% (MAE), 52.9% (MSE), 53.3% (HMAE), and 71.6% (HMSE); better than those of the EGARCH model. That is, the prediction performance of the LSTM was the best of the single models.

When comparing the results of the single GARCH hybrid models based on the DFN and those of a single model, as presented in Table 4, the errors of the hybrid models (G-DFN, E-DFN, and W-DFN) were smaller than in the single models, such as GARCH, EGARCH, EWMA, and DFN, in terms of MAE, MSE, HMAE, and HMSE. However, the prediction errors of the hybrid models based on the DFN were larger than those of the LSTM-based hybrid models. Therefore, we determined that the LSTM can effectively learn temporal patterns of time-series data and characteristics of the long memory phenomenon.

Table 4 indicates that the E-DFN model performed best out of the DFN-based hybrid models. The E-DFN model was created from EGARCH, which also demonstrated the best performance out of the financial time-series models. Although architectures of neuronal networks and the entire data period are different, the prediction error of the model combining the EGARCH and the neural network was also the smallest in the study of Roh (2007),

Table 3
Input variables of the models used in this study.

Input Variables	LSTM (DFN)	G-LSTM (DFN)	E-LSTM (DFN)	W-LSTM (DFN)	GE- LSTM	GW- LSTM	EW- LSTM	GEW- LSTM (DFN)
Explanatory variables	O	O	O	O	O	O	O	O
GARCH param.(2)	X	O	X	X	O	O	X	O
EGARCH param.(3)	X	X	O	X	O	X	O	O
EWMA param.(2)	X	X	X	O	X	O	O	O

Table 4
Results of out-of-sample forecasts with various loss functions.

		Model	MAE	MSE	HMAE	HMSE
Single model		EWMA	0.06559	0.00936	1.82616	3.87996
		GARCH	0.05925	0.00693	1.56660	2.87190
		EGARCH	0.05898	0.00641	1.11166	1.55857
		DFN	0.02083	0.00570	0.65977	0.78589
		LSTM	0.01498	0.00302	0.51864	0.44231
DFN-based hybrid model	with one GARCH-type	W-DFN	0.01943	0.00495	0.63632	0.77037
		G-DFN	0.01941	0.00429	0.60864	0.60309
		E-DFN	0.01701	0.00350	0.56996	0.45170
LSTM-based hybrid model	with one GARCH-type	W-LSTM	0.01322	0.00279	0.49125	0.32835
		G-LSTM	0.01271	0.00252	0.45258	0.29479
		E-LSTM	0.01191	0.00216	0.44651	0.28760
	with two GARCH-type	GW-LSTM	0.01156	0.00160	0.44188	0.28379
		EW-LSTM	0.01136	0.00158	0.43396	0.25292
		GE-LSTM	0.01089	0.00150	0.43279	0.23519
	with three GARCH-type	GEW-LSTM	0.01069	0.00149	0.42911	0.23492

Table 5
Diebold–Mariano (DM) and Wilcoxon signed rank (WS) equal forecast accuracy tests: values above the diagonal are p-values for the DM test and values below the diagonal are p-values for the WS test and boldface values indicate when the p-value is less than 0.05

		G-DFN	E-DFN	W-DFN	G-LSTM	E-LSTM	W-LSTM	GE-LSTM	GW-LSTM	EW-LSTM	GEW-LSTM
G-DFN	MAE		0.58	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	MSE		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	HMAE		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	HMSE		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
E-DFN	MAE	0.68		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	MSE	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	HMAE	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	HMSE	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
W-DFN	MAE	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	MSE	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	HMAE	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	HMSE	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
G-LSTM	MAE	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	MSE	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	HMAE	**0.00**	**0.00**	**0.00**		0.16	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	HMSE	**0.00**	**0.00**	**0.00**		**0.01**	**0.00**	**0.00**	**0.01**	**0.00**	**0.00**
E-LSTM	MAE	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	MSE	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	HMAE	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	0.06	**0.00**	**0.00**
	HMSE	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	0.26	**0.00**	**0.00**
W-LSTM	MAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**
	MSE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**
	HMAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**
	HMSE	**0.00**	**0.00**	**0.00**	0.89	**0.00**		**0.00**	**0.00**	**0.00**	**0.00**
GE-LSTM	MAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	**0.00**
	MSE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	0.51
	HMAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	0.54	**0.02**
	HMSE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**	0.55
GW-LSTM	MAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**
	MSE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**
	HMAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**
	HMSE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**	**0.00**
EW-LSTM	MAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**
	MSE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**
	HMAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**		**0.00**
	HMSE	**0.00**	**0.00**	**0.00**	**0.00**	0.07	**0.00**	**0.00**	**0.00**		**0.00**
GEW-LSTM	MAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.01**	**0.00**	**0.00**	
	MSE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	
	HMAE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	
	HMSE	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	

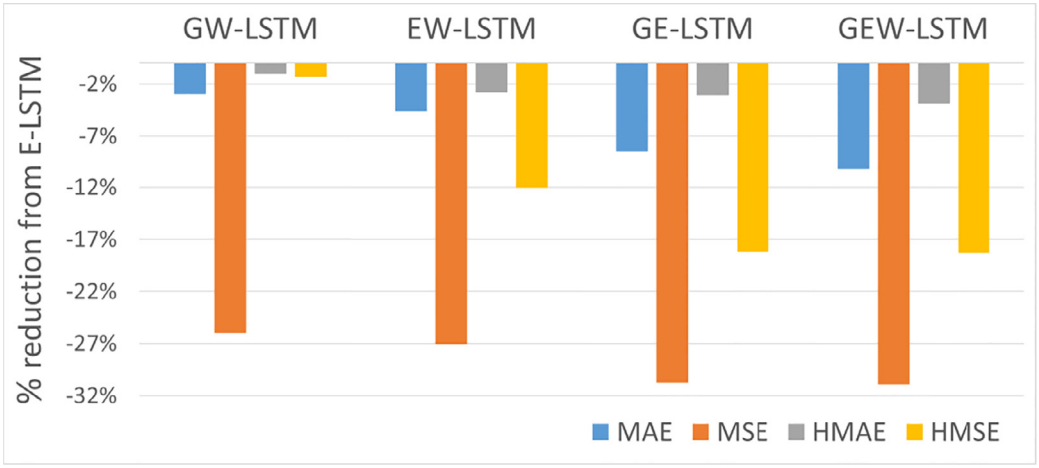

Fig. 7. Comparison of prediction errors of hybrid models combining the LSTM with two or three GARCH models with those of the best-performing LSTM-based single GARCH hybrid model (E-LSTM).

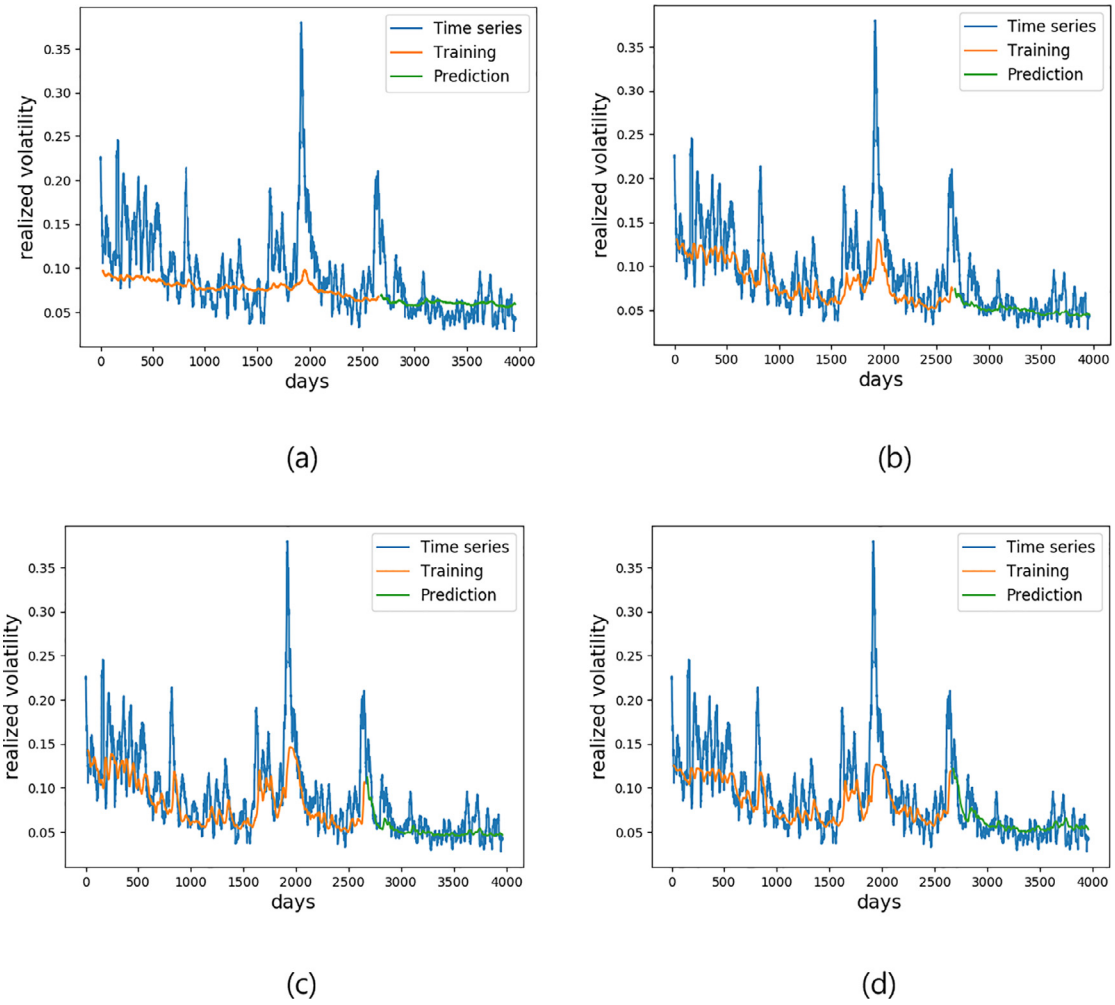

Fig. 8. A comparison of realized volatility forecast results: HMAE loss function is used. (a) E-DFN, (b) E-LSTM, (c) GE-LSTM, and (d) GEW-LSTM.

which predicted the realized volatility of the KOSPI 200 index. Among the LSTM-based hybrid models, the E-LSTM, which was also created from EGARCH, had the lowest MAE, MSE, HMAE, and HMSE values. Among the GARCH-type models, the W-DFN had the lowest performance. The W-DFN was created from the EWMA, which also displayed the worst performance of the DFN hybrid models; these characteristics were also observed for the LSTM-based hybrid models. The results indicate that when a hybrid model is created, the performance of the GARCH-type model that was combined with the neural network affects the performance of the hybrid model. In addition, we found that the neural network models effectively learned the features used to predict volatility from the GARCH-type model parameters used as input.

Table 6
Comparison of out-of-sample forecasts errors over different time horizons for various days-ahead prediction of volatility: Errors are calculated by HMAE

	1 day ahead prediction Windows length (days)			14 days ahead prediction Windows length (days)			21 days ahead prediction Windows length (days)		
	7	15	22	7	15	22	7	15	22
DFN	0.6825	0.6764	0.6598	0.7056	0.6907	0.6753	0.7177	0.7009	0.6770
LSTM	0.5559	0.5289	0.5186	0.5617	0.5331	0.5270	0.5889	0.5514	0.5474
W-DFN	0.6651	0.6410	0.6363	0.6860	0.6673	0.6504	0.6953	0.6892	0.6629
G-DFN	0.6326	0.6130	0.6086	0.6541	0.6362	0.6204	0.6837	0.6672	0.6450
E-DFN	0.6111	0.5878	0.5700	0.6346	0.6122	0.6080	0.6524	0.6323	0.6282
W-LSTM	0.5096	0.5040	0.4912	0.5572	0.5139	0.5100	0.5722	0.5508	0.5108
G-LSTM	0.5046	0.5001	0.4526	0.5511	0.5095	0.4752	0.5629	0.5393	0.5055
E-LSTM	0.5014	0.4633	0.4465	0.5355	0.4819	0.4523	0.5363	0.4861	0.4715
GW-LSTM	0.4990	0.4459	0.4419	0.5249	0.4485	0.4458	0.5331	0.4485	0.4472
EW-LSTM	0.4653	0.4375	0.4340	0.4851	0.4398	0.4381	0.4919	0.4399	0.4398
GE-LSTM	0.4498	0.4331	0.4328	0.4637	0.4375	0.4357	0.4774	0.4395	0.4389
GEW-LSTM	0.4359	0.4305	0.4291	0.4509	0.4364	0.4343	0.4597	0.4376	0.4349

Our comparison of the DFN-based and LSTM-based hybrid models with one GARCH-type model revealed that although the E-DFN model had the best performance of the DFN-based models, and E-LSTM had the best performance of the LSTM-based models, the E-LSTM model experienced about 30%, 38%, 22%, and 36% improvements from the E-DFN model in terms of MAE, MSE, HMAE, and HMSE, respectively. Furthermore, E-LSTM approximately exhibited 39% (MAE), 56% (MSE), 30% (HMAE), and 63% (HMSE) differences in performance compared to W-DFN. Overall, when comparing the DFN and LSTM-based hybrid models, the prediction error of the LSTM hybrid model is markedly lower than the other DFN hybrid models, as shown in Table 4. Even the W-LSTM model, which had the worst performance out of the LSTM hybrid models, had MAE, MSE, HMAE, and HMSE values of 0.01322, 0.00279, 0.49125, and 0.32835, respectively, and showed improvements of 22.3%, 20.3%, 13.8%, and 27.3% over the E-DFN, which is the best model among the DFN-based hybrid models.

Table 4 also shows the comparison results of the models integrating an LSTM with two or more GARCH-type models. Fig. 7 shows how the prediction errors of the hybrid models that integrated two or more GARCH-type models are reduced compared to the best-performing model (E-LSTM) among LSTM-based single GARCH hybrid models. As Table 4 and Fig. 7 demonstrate, the errors of the model combining two or more GARCH-type models with LSTM decreased compared to the single hybrid models. The prediction errors of the poorest-performing model (GW-LSTM) among hybrid LSTM models combining two or more GARCH-type models are smaller than those of the best-performing model (E-LSTM) among LSTM-based single GARCH hybrid models. Among the hybrid models that integrated two or more GARCH-type models, the models with EGARCH demonstrated higher predictive power because their MAE, MSE, HMAE, and HMSE values were smaller than the models without EGARCH. More specifically, among hybrid models with two GARCH-type models, GE-LSTM has the lowest prediction errors, and compared to E-LSTM, GE-LSTM reduced about 8.6% for MAE, 30.7% for MSE, 3% for HMAE, and 18.2% for HMSE. In addition, the GEW-LSTM model combining the three GARCH-type models had lower errors than did the double hybrid models such as GE-LSTM, GW-LSTM, and EW-LSTM. Compared to E-LSTM, the prediction performance of the GEW-LSTM model improved by about 10.2%, 30.9%, 3.9%, and 18.3% for MAE, MSE, HMAE, and HMSE, respectively. We found that the GEW-LSTM was not vastly better than the double hybrid models as certain parameter characteristics overlapped between models. However, the GEW-LSTM exhibited slightly higher predictive performance. As we described in Section 2.4, this result is consistent with our conjecture that creating a hybrid model with various GARCH-type models would improve predictive performance rather than generating a hybrid model with only one GARCH-type model because the economic characteristics captured by each parameter of the GARCH-type models are different.

The results of the DM and WS equal forecast accuracy tests for DFN-based hybrid models and LSTM-based hybrid models are shown in Table 5. The values above the diagonal in Table 5 are p-values for the DM test and values below the diagonal are p-values for the WS test; the boldface values indicate when the p-value is less than 0.05. The null hypothesis of the DM and WS tests is that the null hypothesis assumes that the two predictive models have the same level of accuracy. Thus, if the p-value is less than 0.05, we reject the null hypothesis, that is, we can say that the predictive accuracy of the two competing models is significantly different. If the p-value is greater than 0.05, the null hypothesis cannot be rejected at significance levels of 5% or lower. As shown in Table 5, the null hypothesis can be rejected in almost all cases (314 cases out of a total 324 cases). In both the DM test and the WS test, when the p-value is greater than 0.05, only the results of MAE in the E-DFN and G-DFN models can be compared. Thus, the comparison of the out-of-sample forecasts of the models in this study is statistically significant.

Fig. 8 demonstrates the volatility predicted by the hybrid model versus the realized volatility, which is the target value in our research. In Fig. 8, the models are trained with the HMAE loss function. Panel (a) of Fig. 8 illustrates the predictive results of E-DFN, the best-performing model among the DFN-based single hybrid models; (b) illustrates E-LSTM, the best-performing model among the single LSTM hybrid models; (c) illustrates GE-LSTM, the best-performing model among the double GARCH hybrid LSTM models; and (d) illustrates GEW-LSTM. In addition to the MAE, the graphs confirm that hybrid models combining two or more GARCH-type models have much better predictive performance than E-DFN and E-LSTM do.

In addition, we experimented with three different time windows (7, 15, and 22 days) to analyze non-linear behavior according to different time horizons. Except for the length of the window, all experimental procedures were performed in the same manner as described in Section 3. The measure was also analyzed based on HMAE, which is known to be more suitable for non-linear models. In addition, we conducted three different days ahead (1, 14, and 21 days) prediction experiments to analyze how the forecasting results vary according to the forecasting time points. The experimental results are summarized in Table 6. The GEW-LSTM models have the smallest errors in all cases even with various time horizons (7, 15, and 22 days) and various prediction time points (1, 14, and 21 days ahead). Further, we can see that the longer the window

length, the smaller the out-of-sample forecast error. More specifically, in the case of the 21 days ahead prediction, compared to the error of the 7-window length GEW-LSTM model, errors of the 15-window length GEW-LSTM and 22-window length GEW-LSTM are 4.81% and 5.39% smaller, respectively. Similarly, in both 1 day and 14 days ahead prediction cases, the 22-window length models have the smallest errors although there is a difference in degree. Moreover, the error increases as the prediction time lengthens. Based on the results of the GEW-LSTM model in the 7-window length case, the prediction error of the 14 days ahead model increased by 3.33% and the prediction error of the 21 days ahead model increased by 5.46%. Table 6 shows similar results for the 15- and 22-window length cases.

5. Conclusions

This study proposed a new method to integrate deep neural networks with econometric models. The hybrid models previously proposed by other researchers (Roh, 2007; Wang, 2009; Hajizadeh et al., 2012; Kristjanpoller et al., 2014; Kristjanpoller & Minutolo, 2016; Hernández, 2017) were expanded to combine several econometric models, namely, GARCH-type models, with neural networks. The assumptions of this study are as follows. It is possible to acquire various economic characteristic information, such as the magnitude of volatility shock and the persistence of volatility, in the GARCH (1, 1) model, the direction of volatility impact, the magnitude of volatility impact reflecting the leverage effect, and the persistence of volatility reflecting the leverage effect in the EGARCH model and the persistence of the volatility reflecting the short-term movements and the magnitude of volatility shock reflecting the short-term movements in the EWMA. If the information is input into the LSTM, since the LSTM is exceptional at learning high-level temporal patterns in the time-series data by itself, as the amount of information increases, the features necessary for prediction of the realized volatility can be learned effectively, thereby improving the predictive accuracy. To experimentally verify this, we compared the performance of hybrid models combining a single neural network with a single GARCH-type model and hybrid neural networks combining multiple GARCH models by testing these models' performance on four difference measures (MAE, MSE, HMAE, and HMSE), using them to predict the realized volatility of KOSPI 200 index data. Previous studies (Roh, 2007; Wang, 2009; Hajizadeh et al., 2012) conducted modeling with a single hidden layer feedforward network. However, this study created a hybrid model using the LSTM, which can learn long-range dependency and deeper feedforward neural networks that can efficiently learn more complicated patterns than shallow neural networks.

Consequently, the neural network models combining two or more multiple GARCH-type models exhibited significantly improved prediction performance over hybrid neural networks combining a single GARCH-type model. Compared to the E-LSTM, which had the best predictive performance among single hybrid model neural networks, the GE-LSTM improved performance by 8.6%, 30.7%, 3.1%, and 18.2% for MAE, MSE, HMAE, and HMSE, respectively, and had the best predictive performance among double hybrid models. Moreover, the performance of the GEW-LSTM, which was created by combining three GARCH-type models, improved by 10.2%, 30.9%, 3.9%, and 18.3% for MAE, MSE, HMAE, and HMSE, respectively. From the results of combining two and three models, we observed that the hybrid model combining three GARCH models showed a slight improvement in prediction performance. In other words, we confirmed that the out-of-sample prediction error of the GEW-LSTM model was the lowest for all the measures, MAE, MSE, HMAE, and HMSE. The prediction results for various lengths of window (7, 15, and 22 days) and prediction time points (1, 14, and 21 days) also have the smallest HMAE in the GEW-LSTM and the shorter the window length and the longer the prediction time, the larger the error is. In addition, when comparing the DFN-based hybrid models and the LSTM-based hybrid models in terms of out-of-sample forecast errors, the latter has significantly lower prediction errors than the former. Comparing the E-DFN and E-LSTM, the best model of each single hybrid model, the E-LSTM is better by 30% (MAE), 38% (MSE), 22% (HMAE), and 36% (HMSE). The LSTM is a single model but the prediction error is smaller than the DFN-based integrated model used in previous studies. Moreover, the EGARCH model had the best volatility prediction performance among the GARCH-type models, followed by GARCH and EWMA. The order of performance for hybrid models was identical to the order of GARCH models.

This study used only 7 to 14 financial numerical inputs. However, much more information can be used to predict volatility in various forms such as text or images. For example, news articles, SNS comments, changes in overseas indexes, and various macroeconomic variables can help determine volatility. We expect our volatility predictions to improve if we use more diversified information as inputs and if we were to find an optimized neural network architecture for such input. This study predicted volatility by adding the parameters of financial time-series models as input for the neural network. However, we posit that adding non-quantifiable data could improve predictions using the new multimodal hybrid model. As markets become increasingly complex, the variables to consider in volatility predictions are increasing in various forms. Our next task involves effectively addressing these variables to create better market volatility predictions.

Financial support

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2017R1C1B5018038). The funders had no role in the study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Conflict of interest

None declared.

References

Andersen, T. G., & Bollerslev, T. (1998). Answering the skeptics: Yes, standard volatility models do provide accurate forecasts. *International Economic Review*, 885–905.

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Ebens, H. (2001a). The distribution of realized stock return volatility. *Journal of Financial Economics, 61*(1), 43–76.

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. (2001b). The distribution of realized exchange rate volatility. *Journal of the American Statistical Association, 96*(453), 42–55.

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. (2003). Modeling and forecasting realized volatility. *Econometrica, 71*(2), 579–625.

Baruník, J., & Křehlík, T. (2016). Combining high frequency data with non-linear models for forecasting energy market volatility. *Expert Systems with Applications, 55*, 222–242.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics, 31*(3), 307–327.

Bollerslev, T. (1987). A conditionally heteroskedastic time series model for speculative prices and rates of return. *The Review of Economics and Statistics*, 542–547.

Chen, K., Zhou, Y., & Dai, F. (2015). A LSTM-based method for stock returns prediction: A case study of China stock market. In *Proceedings of the 2015 IEEE international conference on big data (Big Data)* (pp. 2823–2824). IEEE.

Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics, 7*(2), 174–196.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems, 2*(4), 303–314.

Dickey, D. A., & Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American statistical association, 74*(366a), 427–431.

Francis, X., & Roberto, S. (1995). Comparing Predictive Accuracy. *Journal of Business & Economic Statistics, 13*(3).

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica: Journal of the Econometric Society*, 987–1007.
Fuertes, A. M., Izzeldin, M., & Kalotychou, E. (2009). On forecasting daily stock volatility: The role of intraday information and market conditions. *International Journal of Forecasting, 25*(2), 259–281.
Gonzalez Miranda, F., & Burgess, N. (1997). Modelling market volatilities: the neural network perspective. *The European Journal of Finance, 3*(2), 137–157.
Hajizadeh, E., Seifi, A., Zarandi, M. F., & Turksen, I. B. (2012). A hybrid modeling approach for forecasting the volatility of S&P 500 index return. *Expert Systems with Applications, 39*(1), 431–436.
Hamid, S. A., & Iqbal, Z. (2004). Using neural networks for forecasting volatility of S&P 500 Index futures prices. *Journal of Business Research, 57*(10), 1116–1125.
Hamilton, J. D. (1994). *Time series analysis*: 2. Princeton: Princeton university press.
Hernández, E. (2017). Volatility of main metals forecasted by a hybrid ANN-GARCH model with regressors. *Expert Systems with Applications, 84*, 290–300.
Hillebrand, E., & Medeiros, M. C. (2010). The benefits of bagging for forecast models of realized volatility. *Econometric Reviews, 29*(5-6), 571–593.
Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation, 9*(8), 1735–1780.
Hsu, M. W., Lessmann, S., Sung, M. C., Ma, T., & Johnson, J. E. (2016). Bridging the divide in financial market forecasting: machine learners vs. financial economists. *Expert Systems with Applications, 61*, 215–234.
Kim, K. J. (2006). Artificial neural networks with evolutionary instance selection for financial forecasting. *Expert Systems with Applications, 30*(3), 519–526.
Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
Kristjanpoller, W., Fadic, A., & Minutolo, M. C. (2014). Volatility forecast using hybrid neural network models. *Expert Systems with Applications, 41*(5), 2437–2442.
Kristjanpoller, W., & Minutolo, M. C. (2016). Forecasting volatility of oil price using an artificial neural network-GARCH model. *Expert Systems with Applications, 65*, 233–241.
Maknickienė, N., & Maknickas, A. (2012, May). Application of neural network for forecasting of exchange rates and forex trading. In *Proceedings of the 7th international scientific conference on business and management* (pp. 10–11).
Markowitz, H. (1952). Portfolio selection. *The Journal of Finance, 7*(1), 77–91.
Mandelbrot, B. (1967). The variation of some other speculative prices. *The Journal of Business, 40*(4), 393–413.
McAleer, M., & Medeiros, M. C. (2008). Realized volatility: A review. *Econometric Reviews, 27*(1-3), 10–45.
McAleer, M., & Medeiros, M. C. (2011). Forecasting realized volatility with linear and nonlinear univariate models. *Journal of Economic Surveys, 25*(1), 6–18.
Morgan, J. P. (1996). Riskmetrics technical document.
Nelson, D. B. (1991). Conditional heteroskedasticity in asset returns: A new approach. *Econometrica: Journal of the Econometric Society*, 347–370.
Oliveira, N., Cortez, P., & Areal, N. (2017). The impact of microblogging data for stock market prediction: using Twitter to predict returns, volatility, trading volume and survey sentiment indices. *Expert Systems with Applications, 73*, 125–144.
Ormoneit, D., & Neuneier, R. (1996, March). Experiments in predicting the German stock index DAX with density estimating neural networks. In *Proceedings of the IEEE/IAFE 1996 conference on computational intelligence for financial engineering,* (pp. 66–71). IEEE.
Park, H., Kim, N., & Lee, J. (2014). Parametric models and non-parametric machine learning models for predicting option prices: Empirical comparison study over KOSPI 200 Index options. *Expert Systems with Applications, 41*(11), 5227–5237.
Park, J. I., Lee, D. J., Song, C. K., & Chun, M. G. (2010). TAIFEX and KOSPI 200 forecasting based on two-factors high-order fuzzy time series and particle swarm optimization. *Expert Systems with Applications, 37*(2), 959–967.
Roh, T. H. (2007). Forecasting the volatility of stock price index. *Expert Systems with Applications, 33*(4), 916–922.
Ryu, D. (2012). Implied volatility index of KOSPI200: Information contents and properties. *Emerging Markets Finance and Trade, 48*(sup2), 24–39.
Shim, H., Kim, M. H., & Ryu, D. (2017). Effects of intraday weather changes on asset returns and volatilities. *Zbornik radova Ekonomskog fakulteta u Rijeci: časopis za ekonomsku teoriju i praksu, 35*(2), 301–330.
Tseng, C. H., Cheng, S. T., Wang, Y. H., & Peng, J. T. (2008). Artificial neural network model of the hybrid EGARCH volatility of the Taiwan stock index option prices. *Physica A: Statistical Mechanics and its Applications, 387*(13), 3192–3200.
Wang, Y. H. (2009). Nonlinear neural network forecasting model for stock index option price: Hybrid GJR–GARCH approach. *Expert Systems with Applications, 36*(1), 564–570.
Wilhelmsson, A. (2006). GARCH forecasting performance under different distribution assumptions. *Journal of Forecasting, 25*(8), 561–578.
Yao, Y., Zhai, J., Cao, Y., Ding, X., Liu, J., & Luo, Y. (2017). Data analytics enhanced component volatility model. *Expert Systems with Applications, 84*, 232–241.