A convolutional neural network based approach to financial time series prediction

Dr. M. Durairaj1 · B. H. Krishna Mohan1

Received: 30 June 2021 / Accepted: 24 February 2022 / Published online: 23 March 2022
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

Abstract
Financial time series are chaotic that, in turn, leads their predictability to be complex and challenging. This paper presents a novel financial time series prediction hybrid that involves Chaos Theory, Convolutional neural network (CNN), and Polynomial Regression (PR). The financial time series is first checked in this hybrid for the presence of chaos. The chaos in the series of times is later modeled using Chaos Theory. The modeled time series is input to CNN to obtain initial predictions. The error series obtained from CNN predictions is fit by PR to get error predictions. The error predictions and initial predictions from CNN are added to obtain the final predictions of the hybrid model. The effectiveness of the proposed hybrid (Chaos + CNN + PR) is tested by using three types of Foreign exchange rates of financial time series (INR/USD, JPY/USD, SGD/USD), commodity prices (Gold, Crude Oil, Soya beans), and stock market indices (S&P 500, Nifty 50, Shanghai Composite). The proposed hybrid is superior to Auto-regressive integrated moving averages (ARIMA), Prophet, Classification and Regression Tree (CART), Random Forest (RF), CNN, Chaos+CART, Chaos+RF and Chaos+CNN in terms of MSE, MAPE, Dstat, and Theil’s U.

Keywords Deep learning · Time series prediction · CNN · Chaos · Polynomial regression · Exchange rate · Stock market index · Commodity price

1 Introduction

The Financial Time Series is a collection of observations of Financial Variable(s) recorded regularly. E.g., daily exchange rates, daily stock market index values, and daily commodity prices are financial time series. In general, The financial time series is chaotic and noisy [39]. A chaotic time series is not linear and sensitive to initial conditions [7]. Financial Time series are also noisy, and their statistical properties vary with time. This property makes the prediction impossible [11, 19]. Building the right prediction model that can capture nonlinearity present in the time series is always challenging. It reveals, therefore, that the prediction of financial time series is a difficult and complex task.

Several researchers have demonstrated that an ensemble or hybrid forecasting model for time series can perform better in comparison with stand-alone forecasting models [4, 32]. A hybrid combines two or more stand-alone forecasting models into a mixed model to improve prediction accuracy and overcome the deficiencies of stand-alone models.

Chaos theory [26, 38] models nonlinear financial time series by using lag and embedding dimension in which a lag is the time delay, and embedding dimension is the number of variables required to capture the nonlinear dynamics of financial time series.

Applying deep learning approaches can help achieve better prediction accuracy [3, 6]. Deep learning, a subset of machine learning, allows Artificial Neural Networks (ANNs) to learn multi-level abstraction data representations (hierarchical learning) [10, 16]. The ANNs can construct a nonlinear and complex function that maps inputs to output. These are applied to solve various financial problems such as prediction of stock markets, optimization of...
portfolios, processing, and execution of trade information [43]. This field is still relatively unexplored, however.

A CNN [14] is a special case of the neural network that consists of one or more convolutional layers, often with a subsampling layer, which are followed by one or more fully connected layers as in a standard neural network. The CNNs are a type of neural network developed for two-dimensional image data. However, they can be used for one-dimensional data such as sequences of text and time series [15].

This paper presents a hybrid model involving Chaos Theory, CNN, and PR to predict financial time series as follows. The financial time series in this hybrid is checked for chaos. The chaotic modeled time series is input to CNN to obtain initial predictions. The error series obtained from CNN predictions is fit by PR to obtain error predictions. To get final forecasts from the hybrid model, CNN error predictions and initial predictions are added. Our goal is to build a more accurate model to predict different financial time series such as exchange rates, commodity prices and stock prices.

Though there are Chaos-based hybrids, such as Chaos+MLP+PSO [27], Chaos+MLP+MPSO and Chaos+MLP+NSGA-II [31], present in the literature (see Table 1), the second-stage of the approaches modeling error series aforementioned are complex and time consuming as there are more parameters to be tuned. So, we used a simple PR to model error series as it can capture nonlinearity present in error series very well. In addition, no approach is comprehensively tested for its efficacy on three types of financial time series.

The contributions of this paper include:

- Two novel chaotic hybrids, Chaos+CNN and Chaos+CNN+PR, are proposed for prediction over 30 years of financial data.
- Solutions to three different financial time series prediction problems, including predicting exchange rates, predicting stock index, and predicting commodity prices.
- Comparative study of proposed hybrids with stand-alone time series prediction models including ARIMA, Prophet, CART, RF and CNN.
- Comparative study of proposed hybrids with other related chaos-based hybrids such as Chaos+CART [28] and Chaos+RF [28] found in literature.

The remainder of the paper is arranged accordingly: The related literature is presented by Sect. 2. Subsequently, Sect. 3 describes in detail the approach proposed. Next, Sect. 4 describes the experimental design, and Sect. 5 discusses the results. Finally, the paper is concluded.

### 2 Related literature

There are numerous hybrids for Time series in financial literature and are summarized in Cavalcante et al. [6], Huang et al. [13], Pfeiffer and Hohmann [25], Mochoñ et al. [22], Li and Ma [17], Bahrammirzaee [2], and Pradeepkumar and Ravi [30]. The deep learning hybrids for financial time series prediction are also found in last two decades of literature and are recently well summarized by Durairaj and Mohan [9] and deep learning approaches for financial time series forecasting are reviewed by Sezer et al. [34].

#### 2.1 CNN-based hybrids

This section presents various related CNN-based hybrids and chaos-based hybrids proposed for financial time series prediction connected with the works mentioned above. The CNN-based hybrids are as follows:

Livieris et al. [18] proposed a CNN–LSTM model for gold price time series forecasting in which CNN is used for learning an internal representation of time series and Long

| Year | Author(s) | Chaos-based hybrids |
|------|-----------|---------------------|
| 2003 | Pavlidis et al. [24] | Chaos theory hybrid methodology, ANN, Cluster, and PSO/DE |
| 2010 | Huang et al. [12] | Chaos + SVR |
| 2014 | Pradeepkumar and Ravi [27] | Chaos+ANN+PSO*, Chaos+PSO+ANN* |
| 2016 | Pradeepkumar and Ravi [28] | Chaos+QRRF*, Chaos+QR, Chaos+RF |
| 2017 | Pradeepkumar and Ravi [29] | Chaos+CART, Chaos+CART-EB, Chaos+TreeNet, Chaos+LASSO, Chaos+RFTE, Chaos+MARS* |
| 2017 | Ravi et al. [31] | Chaos+MLP+MOPSO, Chaos+MLP+NSGA-II* |

ANN artificial neural network, QR quantile regression, QRRF quantile regression random forest, CART-EB CART ensemble, RFTE RF tree ensemble, PSO particle swarm optimization, DE differential evolution, MOPSO multi-objective PSO, MARS multivariate adaptive regression splines, LASSO least absolute shrinkage selection operator, NSGA-II non-dominated sorting genetic algorithm-II

*Winner Hybrid
Short Term Memory (LSTM) is used for identifying short-term and long-term dependencies. Similarly, Vidal and Kristjanpoller [37] proposed another CNN-LSTM hybrid model, which could include images as input which provides a wide variety of information associated with both static and dynamic characteristics of the series. The authors utilized this approach for predicting gold price volatility. Selvin et al. [33] applied a sliding window approach and proposed a new CNN-based hybrid, namely the CNN-Sliding Window model, in which a sliding window is used for predicting future values on a short-term basis.

### 2.2 Chaos-based hybrids

Table 1 presents the Hybrids based on chaos theory found in the literature to predict financial time series. All of these concluded that the proposed chaos-based hybrids outperformed stand-alone models.

### 3 Proposed approach

In the proposed hybrid, a financial time series is checked for the presence of chaos. Lyapunov exponent [31] is used for this purpose. Chaos theory is then employed to build the scalar time series phase space [23, 35]. Optimum lag and optimal dimensional values are required for building phase space. Akaike Information Criterion (AIC) [1] It is used for optimal time series lag selection. Method of Cao’s [5] is used for the optimal dimensions of embedding. Once optimal lag and optimal embedding dimension are obtained from time series, phase space can be reconstructed using Chaos Theory. Later, CNN is used for obtaining initial predictions, and finally, PR is used to fine-tune predictions. The proposed hybrid is compared with ARIMA [21], Prophet (https://facebook.github.io/prophet/), CNN, CART, RF, Chaos+CART [28], Chaos+RF [28] and Chaos+CNN.

Table 2 presents the notations along with their interpretations used in the proposed approach.

The proposed hybrid approach is described as follows. Let \( Y = \{y_1,y_2,y_3,\ldots,y_k,y_{k+1},\ldots,y_N\} \) be a time series with \( N \) Comments sometimes recorded \( t = \{1,2,3,\ldots,k,k+1,\ldots,N\} \). Then perform the following:

1. For chaos to occur, check \( Y \). When there is chaos, get optimum lag (\( l \)) and optimum embedding dimensions (\( m \)) from \( Y \).
2. Once optimal lag and embedding dimension values are obtained, reconstruct phase space from \( Y \).
3. After phase space is reconstructed, partition \( Y \) into \( Y_{Train} = \{y_i; \ t = lm + 1, lm + 2, \ldots,k\} \) and \( Y_{Test} = \{y_i; \ t = k + 1, k + 2, \ldots,N\} \).
4. Input \( Y_{Train} \) to CNN, train CNN to get initial predictions of training set using Eq. 1.

\[
y_t^f = f_1(y_{t-l}, y_{t-2l}, \ldots, y_{t-ml})
\]

where \( t = lm + 1, lm + 2, \ldots,k \)

5. Obtain initial test set predictions by input \( Y_{Test} \) to trained CNN by replacing \( t = \{k + 1, k + 2, \ldots,N\} \) in Eq. 1.

6. Compute training set of prediction errors using Eq. 2 and test set of prediction errors by replacing \( t = \{k + 1, k + 2, \ldots,N\} \) in Eq. 2.

\[
e_t = y_t - y_t^f
\]

where \( t = lm + 1, lm + 2, \ldots,k \)

7. Fit Polynomial Regression to training set of errors and obtain training set error predictions using Eq. 3. Similarly fit PR to test set of errors and obtain test set error predictions by replacing \( t = \{k + 1, k + 2, \ldots,N\} \) in Eq. 3.

\[
e_t^f = f_2(e_t)
\]

where \( t = lm + 1, lm + 2, \ldots,k \)

8. Add training set initial predictions and training set error predictions to obtain final training set predictions using Eq 4. Similarly, add test set initial predictions and test set error predictions to obtain final test set predictions by replacing \( t = \{k + 1, k + 2, \ldots,N\} \) in Eq. 4.

\[
y_t = y_t^f + e_t^f
\]

where \( t = lm + 1, lm + 2, \ldots,k \)

| Table 2 | Notations used in proposed approach |
|----------|-----------------------------------|
| Notation | Interpretation                     |
| \( l \)  | Optimal lag                       |
| \( m \)  | Optimal embedding dimension       |
| \( y_t \) | Actual observation at time \( t \) |
| \( e_t \) | Error in time achievement \( t \) |
| \( e_t^f \) | Error prediction in due course \( t \) |
| \( y_t^f \) | Prediction at the beginning time \( t \) |
| \( y_t^e \) | Time to finish prediction \( t \) |
| \( f_1(.) \) | Nonlinear function used by CNN to obtain predictions |
| \( f_2(.) \) | Linear function used by PR to obtain predictions |
4 Experimental design

4.1 Datasets used

Various Datasets are used in this paper to observe the effectiveness of proposed hybrids. These daily datasets of 30 years approximately include:

- Three exchange rates are collected from the Federal Reserve: Indian Rupees (INR)/USD, Japanese Yen (JPY)/USD, Singapore Dollar (SGD)/USD.
- The Composite Index of Investing.com collects three stock market indicators, Standard & Poor (S&P) 500, Nifty 50, and Shanghai.
- Three commodity prices in US Dollars namely Crude Oil Price, Gold Price, and Soyabeans price are collected from Investing.com.

Table 3 presents these datasets along with corresponding dates, number of observations, training set, and test set. Here, the financial time series prediction problem is modeled as a supervised learning problem. Thus, each dataset is divided into a training set (80%) and a test set (20%) of observations. First, all of these datasets are checked for chaos, and it is found that chaos is present in each dataset. Later, phase space is reconstructed with the corresponding optimum lag and ideal insertion dimensions from each dataset (Fig. 1).

Table 4 presents various descriptive statistical measures of the datasets such as minimum, mean, median, maximum, standard deviation, skewness, and kurtosis. The prices of Crude Oil (USD) are in the range of (−37.63, 145.29), Gold (USD) are in the range of (253, 2069.4), and Soyabeans (USD) are in the range of (410, 1764.75). The stock prices of Nifty 50 are in the range of (788.15, 14730.95), Shanghai Composite Index are in the range of (104,39,6092.06), and S&P 500 are in the range of (295,45012,3862,959961). The ranges of both commodity prices and stock prices are too much varied because of COVID-19’s impact. The exchange rates of INR/USD are in the range of (16.8, 76.975), JPY/USD are in the range of

![Fig. 1 Architecture of the proposed hybrid](image-url)
Table 4 Descriptive statistics for all datasets

| Data                      | Count | Min  | Mean    | Median  | Max   | SD      | Skewness | Kurtosis |
|---------------------------|-------|------|---------|---------|-------|---------|----------|----------|
| 1. Crude oil price (USD)  |       |      |         |         |       |         |          |          |
| All data                  | 7890  | – 37.63 | 47.70264 | 41.405  | 145.29 | 28.83285 | 0.73005  | – 0.49341 |
| Training set              | 6312  | 10.72 | 46.91822 | 30.35   | 145.29 | 31.72228 | 0.75695  | – 0.81044 |
| Test set                  | 1578  | – 37.63 | 50.84032 | 50.915  | 76.41  | 10.91546 | – 0.63922 | 2.93409  |
| 2. Gold price (USD)       |       |      |         |         |       |         |          |          |
| All data                  | 7907  | 253  | 797.95469 | 465     | 2069.4 | 515.3496 | 0.53530  | – 1.27063 |
| Training set              | 6326  | 253  | 648.62458 | 388.1   | 1888.7 | 457.3733 | 1.17048  | – 0.13432 |
| Test set                  | 1581  | 1070.8 | 1395.46405 | 1324.2  | 2069.4 | 212.11469 | 1.33137  | 0.90218  |
| 3. Soybeans price (USD)   |       |      |         |         |       |         |          |          |
| All data                  | 8063  | 410  | 833.65716 | 775     | 1764.75 | 301.45914 | 0.53530  | – 1.27063 |
| Training set              | 6451  | 410  | 803.83398 | 658.25  | 1764.75 | 327.03820 | 1.17048  | – 0.13432 |
| Test set                  | 1612  | 803.5 | 953.00537 | 936.75  | 1430   | 93.48183 | 1.718141 | 4.81299  |
| 4. Nifty 50 stock price   |       |      |         |         |       |         |          |          |
| All data                  | 7362  | 104.39 | 1994.61469 | 1924.3  | 6092.06 | 1075.78886 | 0.50195  | 0.08418  |
| Training set              | 5890  | 104.39 | 1702.09134 | 1526.139 | 6092.06 | 989.68912 | 1.13437  | 2.17665  |
| Test set                  | 1472  | 2464.36 | 3165.10552 | 3114.73 | 5166.35 | 395.55923 | 1.82079  | 5.53021  |
| 5. Shanghai composite index|     |      |         |         |       |         |          |          |
| All data                  | 7831  | 295.450012 | 1335.66791 | 1210.930054 | 3862.959961 | 757.17977 | 0.93029  | 0.42257  |
| Training set              | 6265  | 295.450012 | 1023.83847 | 1110.469971 | 2032.359985 | 418.18596 | 0.09975  | 0.8005   |
| Test set                  | 1566  | 1833.40002    | 2583.18476 | 2577.915039 | 3862.959961 | 471.20240 | 0.49377  | 0.57165  |
| 6. S&P 500 stock index    |       |      |         |         |       |         |          |          |
| All data                  | 8093  | 16.8  | 46.88723 | 45.5    | 76.975 | 14.00687 | 0.14052  | – 0.49158 |
| Training set              | 6475  | 16.8  | 41.56766 | 43.73   | 68.805 | 10.00020 | – 0.42131 | 0.207695 |
| Test set                  | 1618  | 61.3580 | 68.17536 | 67.41149 | 76.975 | 3.83363 | 0.35588  | 0.88452  |
| 7. INR/USD                |       |      |         |         |       |         |          |          |
| All data                  | 8101  | 75.82 | 110.50040 | 109.98  | 159.88 | 15.10520 | 0.03674  | 0.43326  |
| Training set              | 6481  | 75.82 | 110.26933 | 109.54  | 159.88 | 16.648432 | 0.06609  | 0.09937  |
| Test set                  | 1620  | 99.89 | 111.42484 | 110.62  | 125.62 | 5.5728100 | 0.54998  | 0.33050  |
| 8. JPY/USD                |       |      |         |         |       |         |          |          |
| All data                  | 8101  | 1.2006 | 1.51456   | 1.4703  | 1.9085   | 0.18029 | 0.16609  | 1.246123 |
| Training set              | 6481  | 1.2006 | 1.55079   | 1.5905  | 1.9085   | 0.18393 | 0.26470  | 1.123532 |
| Test set                  | 1620  | 1.2976 | 1.36961   | 1.3644  | 1.4598   | 0.03072 | 0.327661 | 0.215345 |

The skewness measures asymmetry of data. The value Zero indicates the data is perfectly symmetric. The positive value indicates the tail of the distribution is more stretched on the side above mean. The negative value indicates that the tail of the distribution is more stretched on the side below the mean. The tails of the distribution of all commodity prices, stock prices and exchange rates are more stretched on the side above the mean. The Kurtosis characterizes the relative peakedness or flatness of a distribution compared with the normal distribution. Positive kurtosis indicates a relatively peaked distribution and a negative kurtosis indicates a relatively flat distribution. The datasets of all commodity prices, Nifty 50 Stock Price, INR/USD and SGD/USD have relatively flat distribution. The stock prices such as Shanghai Composite Index and S&P 500 and JPY/USD have relatively peaked distribution.

(75.82,109.98), and SGD/USD are in the range of (1.2006,1.9085).
4.2 Tasks performed and tools employed

Various tasks are carried out during the experimentation. Such tasks, as well as the tools used to conduct them, are presented in the Table 5. The Lyapunov Exponent (\(\lambda\)) is used to check for chaos, the AIC is used to achieve optimal lag, and Cao’s technique is used to provide optimal embedding dimension, as shown in Table 5. For additional information on the descriptions of the tasks aforementioned, readers are suggested to refer to [31].

While experimenting with the datasets, various parameters are obtained, and some parameters are utilized in common. Table 6 presents the optimal values for chaotic parameters obtained. \(\lambda \geq 0\) denotes the presence of chaos. From the table, it is clear that all of the datasets have chaos. The optimal chaotic parameters such as lag \(l\) and embedding dimension \(m\) are also presented in Table 6. The estimateEmbeddingDim(.) method from “nonlinearTimeseries” package implemented Cao’s method [5].

The optimal parameters for ARIMA \((p, d, q)\) will be presented in respective sections. The optimal \(p, d,\) and \(q\) values of the ARIMA model are obtained using auto_arima(.) from “pmdarima” module of Python. The commonly used parameters for all datasets are as follows. The CNN architecture used here consists of one fully connected dense layer of 50 nodes. Each node is with the activation function of ReLU. For the CNN to be trained for 500 epochs, adam optimizer is used with MSE as a loss function. It also consists of a convolutional layer and a pooling layer. Scaled values using MinMaxScaler are input to CNN, Chaos+CNN, and Chaos+CNN+PR. While modeling errors using PR, second-degree polynomial regression is used.

| Task                                           | Package/module          | Function/measure/class | Tool used |
|------------------------------------------------|-------------------------|------------------------|-----------|
| Checking for the presence of chaos             | nolds                   | lyap_r(.)              | Python    |
| Finding optimal lag                           | –                       | AIC                    | Gretl     |
| Finding optimal embedding dimension           | nonlinearTseries        | estimateEmbeddingDim(.)| R         |
| Importing data                                | pandas                  | read_csv(.)            | Python    |
| Partitioning data                             | scikit-learn            | train_test_split(.)    | Python    |
| Fitting ARIMA to data                         | statsmodels             | ARIMA(.).fit(), forecast(.)| Python |
| Fitting Prophet to data                       | fbprophet               | Prophet(.).fit(), predict(.)| Python |
| Fitting CNN to data                           | keras                   | CNN(.), predict(.)     | Python    |
| Fitting PR to data                            | scikit-learn            | PolynomialFeatures(.).LinearRegression(.).predict(.)| Python |
| Computing MSE                                 | scikit-learn            | mean_squared_error(.)  | Python    |
| Computing Dstat                               | –                       | –                      | Python    |
| Computing Theil’s \(U\)                      | –                       | –                      | Python    |
| Checking for statistical significance         | forecast                | dm.test(.)             | R         |

4.3 Performance measures used

The suggested hybrid’s performance is measured using four performance measures: Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), Directional Change Statistic (Dstat), and Theil’s Inequality Coefficient (Theil’s \(U\)).

By measuring the average of squared errors, the MSE (see Eq. 5) determines how well the model predicts the response [20]. The MAPE [20] calculates the absolute numbers of errors in percentage terms to determine how well the model predicts the response. An MSE/MAPE score near 0 suggests that the suggested model could produce predictions that are more accurate than the observed data.

Table 6 Chaotic parameters

| Dataset            | \(\lambda\)     | \(l\) | \(m\) |
|--------------------|------------------|------|------|
| Crude oil price    | 0.001618635      | 4    | 9    |
| Gold price         | 0.000222457      | 10   | 8    |
| Soyabeans price    | 0.003601366      | 10   | 8    |
| Nifty 50           | 0.002267289      | 10   | 8    |
| Shanghai composite | 0.003585269      | 8    | 7    |
| S&P 500            | 0.001685243      | 1    | 9    |
| INR/USD            | 0.00099022       | 6    | 10   |
| JPY/USD            | 0.003709193      | 1    | 8    |
| SGD/USD            | 0.002180623      | 2    | 8    |
\[
\text{MSE} = \frac{1}{N} \sum_{t=1}^{N} (y_t - \hat{y}_t)^2
\]
\[
\text{MAPE} = \frac{1}{N} \sum_{t=1}^{N} \left| \frac{y_t - \hat{y}_t}{y_t} \right|
\]

Yao and Tan [39] developed a measure (expressed in percentages) namely Dstat (see Eq. 7) to measure the directional change of time series. Higher the value of Dstat, better the movements of time series are captured by the model.

\[
\text{Dstat} = \frac{1}{N} \sum_{t=1}^{N} a_t \times 100\%
\]
\[
\text{where } a_t = \begin{cases} 1, & \text{if } (y_{t+1} - y_t) \times (\hat{y}_{t+1} - \hat{y}_t) \geq 0 \\ 0, & \text{Otherwise} \end{cases}
\]

Theil’s U indicates how near a projected time series is to the actual time series [20, 36]. The value of U (see Eq. (8)) is usually somewhere between 0 and 1. U = 0 indicates that \( y_t = \hat{y}_t \) for all observations and a perfect fit exists, whereas U = 1 indicates that the performance is poor. A Theil’s U value that is closer to 0 suggests that the suggested model could produce more accurate predictions.

\[
U = \frac{\left( \frac{1}{N} \sum_{t=1}^{N} (y_t - \hat{y}_t)^2 \right)^{1/2}}{\left( \frac{1}{N} \sum_{t=1}^{N} (y_t)^2 \right)^{1/2} + \frac{1}{N} \sum_{t=1}^{N} (\hat{y}_t)^2}
\]

In all of the related equations of these performance measures, \( y_t \) is the actual value at time t, \( \hat{y}_t \) is the predicted value obtained using the proposed approach at time t and N is the number of predicted values.

## 5 Results and discussion

The results of each dataset are described as follows. It is important to note that, for each dataset, the proposed hybrid (Chaos+CNN+PR) is compared with ARIMA, Prophet, CNN, CART, RF, Chaos+CNN, Chaos+RF [28], Chaos+RF [28] and Chaos+CNN in terms of MSE, MAPE, Dstat, and Theil’s U.

### 5.1 INR/USD

The INR/USD test set results of prediction approaches are presented in Table 7. The table reveals that the proposed hybrid, Chaos+CNN+PR, outperformed all other approaches in terms of MSE, MAPE, Dstat and Theil’s U. The performance measures MSE, MAPE and Theil’s U are very much closer to 0 indicate that predictions are very much closer to actual values. Dstat value 100 indicates that directional change is fully captured by the proposed hybrid.

Among the standard prediction approaches (ARIMA (3,1,2), Prophet, CNN, CART, RF), CNN could yield better predictions in terms of MSE, MAPE and Theil’s U. However, it could not capture the direction change better. In this context, CART could perform better.

Similarly, among Chaos-based hybrids (Chaos+CART, Chaos+RF, Chaos+CNN), the novel hybrid, Chaos+CNN, could yield better predictions in terms of MSE, MAPE and Theil’s U. However, it could not capture the direction change better than dominative Chaos+CART.

Figure 2 depicts predictions of the test set of INR/USD. The predictions are obtained from CNN, Chaos+CNN, and Chaos+CNN+PR. From the figure, it can be observed that the predictions obtained using Chaos+CNN+PR are very much closer to actual values. It is also worth noting that the predictions obtained using CNN are better than that of Chaos+CNN.

### 5.2 JPY/USD

Table 8 shows the JPY/USD test set results of prediction techniques. The suggested hybrid, Chaos+CNN+PR, beat all previous techniques in terms of MSE, MAPE, Dstat, and Theil’s U, as shown in the table. The MSE, MAPE, and Theil’s U performance metrics are all extremely close to 0, indicating that predictions are very close to actual values. Dstat value 100 shows that the suggested hybrid completely captures directional change.

Among the conventional prediction techniques (ARIMA (0,1,0), Prophet, CNN, CART, RF), CNN followed by RF could produce superior forecasts in terms of MSE, MAPE, and Theil’s U. However, it fell short of capturing the change in direction. CART may function better in this situation.

Similarly, among Chaos-based hybrids (Chaos+CART, Chaos+RF, Chaos+CNN), the proposed hybrid, Chaos+CNN, could provide superior forecasts in terms of MSE, MAPE, and Theil’s U. It could not, however, record the direction change better than Chaos+CART.

The predictions of the test set of JPY/USD are shown in Fig. 3. CNN, Chaos+CNN, and Chaos+CNN+PR are used to obtain the predictions and they are shown in the figure. The predictions achieved using Chaos+CNN+PR are significantly closer to real values as seen in the figure. It’s also worth mentioning that the forecasts made with CNN are superior to those made using Chaos+CNN.
5.3 SGD/USD

The SGD/USD test set results of prediction approaches are presented in Table 9. The table reveals that the proposed hybrid, Chaos+CNN+PR, outperformed all other approaches in terms of MSE, MAPE, Dstat and Theil’s $U$. The performance measures MSE, MAPE and Theil’s $U$ are very much closer to 0 indicate that predictions are very much closer to actual values. Dstat value 100 indicates that directional change is fully captured by the proposed hybrid.

Among the standard prediction approaches (ARIMA (1,0,0), Prophet, CNN, CART, RF), CNN followed by RF could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better. In this context, CART could perform better.

Similarly, among Chaos-based hybrids (Chaos+CART, Chaos+RF, Chaos+CNN), the novel hybrid, Chaos+CNN, could yield better predictions in terms of MSE, MAPE and
Theil’s $U$. However, it could not capture the direction change better than Chaos+CART.

Figure 4 depicts predictions of the test set of SGD/USD. The predictions are obtained from CNN, Chaos+CNN, and Chaos+CNN+PR. From the figure, it can be observed that the predictions obtained using Chaos+CNN+PR are very much closer to actual values. It is also worth noting that the predictions obtained using CNN are better than that of Chaos+CNN.

### 5.4 S&P 500 stock index

The S&P 500 Stock Index test set results of prediction approaches are presented in Table 10. The table reveals that the proposed hybrid, Chaos+CNN+PR, outperformed all other approaches in terms of MSE, MAPE, Dstat and Theil’s $U$. The performance measures MSE, MAPE and Theil’s $U$ are very much closer to 0 indicate that predictions are very much closer to actual values. Dstat value 100
indicates that directional change is fully captured by the proposed hybrid.

Among the standard prediction approaches (ARIMA (1,1,1), Prophet, CNN, CART, RF), CNN could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better. In this context, CART could perform better.

Similarly, among Chaos-based hybrids (Chaos+CART, Chaos+RF, Chaos+CNN), the novel hybrid, Chaos+CNN, could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better than Chaos+CART.

Figure 5 depicts predictions of the test set of S&P 500 Stock Index. The predictions are obtained from CNN, Chaos+CNN, and Chaos+CNN+PR. From the figure, it can be observed that the predictions obtained using Chaos+CNN+PR are very much closer to actual values. It is also worth noting that the predictions obtained using Chaos+CNN are better than that of CNN.

| Forecasting model | MSE   | MAPE  | DStat  | Theil $U$ |
|-------------------|-------|-------|--------|-----------|
| ARIMA             | 0.024654 | 12.260339 | 51.822112 | 0.007313 |
| Prophet           | 0.049610 | 19.039075 | 51.142680 | 0.015482 |
| CNN               | 2.378308e - 05 | 0.270992 | 50.833848 | 6.345399e - 06 |
| CART              | 4.499225e - 05 | 0.384234 | 55.095738 | 1.198772e - 05 |
| RF                | 2.754348e - 05 | 0.291158 | 48.857319 | 7.339461e - 06 |
| Chaos+CART        | 4.544950e - 05 | 0.381212 | 56.948733 | 1.210897e - 05 |
| Chaos+RF          | 2.695724e - 05 | 0.287179 | 49.351451 | 7.183174e - 06 |
| Chaos+CNN         | 3.842139e - 05 | 0.339434 | 50.216182 | 1.023644e - 05 |
| Chaos+CNN+PR      | 3.175296e - 12 | 0.000130 | 100.0 | 8.459333e - 13 |

Fig. 4 Predictions of proposed hybrid for test set of SGD/USD
5.5 Nifty 50 stock index

The Nifty 50 Stock Index test set results of prediction approaches are presented in Table 11. The table reveals that the proposed hybrid, Chaos+CNN+PR, outperformed all other approaches in terms of MSE, MAPE, Dstat and Theil’s $U$. The MSE value of proposed hybrid is better than the remaining approaches. The MAPE and Theil’s $U$ are very much closer to 0 indicate that predictions are very much closer to actual values. Dstat value 100 indicates that directional change is fully captured by the proposed hybrid.

Among the standard prediction approaches (ARIMA (0,1,1), Prophet, CNN, CART, RF), CNN followed by ARIMA could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better. In this context, CART could perform better.

Similarly, among Chaos-based hybrids (Chaos+CART, Chaos+RF, Chaos+CNN), the novel hybrid, Chaos+CNN,
could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better than Chaos+CART.

Figure 6 depicts predictions of the test set of Nifty 50 Stock Index. The predictions are obtained from CNN, Chaos+CNN, and Chaos+CNN+PR. From the figure, it can be observed that the predictions obtained using Chaos+CNN+PR are very much closer to actual values. It is also worth noting that the predictions obtained using CNN are better than that of Chaos+CNN.

### Table 11 Test set results of Nifty 50 stock index

| Forecasting model | MSE     | MAPE   | DStat  | Theil $U$ |
|-------------------|---------|--------|--------|-----------|
| ARIMA             | 3400655.703089 | 17.260416 | 55.139442 | 0.018258   |
| Prophet           | 849534.677585  | 6.739681  | 52.270916 | 0.003937   |
| CNN               | 16129.754511   | 0.850068  | 50.836653 | 7.499417e-05 |
| CART              | 4511147.427545 | 19.200982 | 87.808764 | 0.024733   |
| RF                | 4188601.943174 | 18.087281 | 87.808764 | 0.022805   |
| Chaos+CART        | 4510898.369836 | 19.194321 | 87.569721 | 0.024733   |
| Chaos+RF          | 4232360.770306 | 18.233875 | 87.888446 | 0.023067   |
| Chaos+CNN         | 1560419.525616 | 10.837223 | 50.677290 | 0.008003   |
| Chaos+CNN+PR      | 2.639324      | 0.016042 | 100.0   | 1.222894e-08 |

5.6 Shanghai composite index

The Shanghai Composite Index test set results of prediction approaches are presented in Table 12. The table reveals that the proposed hybrid, Chaos+CNN+PR, outperformed all other approaches in terms of MSE, MAPE, Dstat and Theil’s $U$. The MSE value of Chaos+CNN+PR is very much better than the remaining approaches. And also, the MAPE and Theil’s $U$ values are very much closer to 0 indicate that predictions are very much closer to actual.

![Fig. 6 Predictions of proposed hybrid for test set of Nifty 50 stock index](image-url)
values. Dstat value 100 indicates that directional change is fully captured by the proposed hybrid.

Among the standard prediction approaches (ARIMA (3,1,3), Prophet, CNN, CART, RF), CNN followed by RF could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better. In this context, CART and ARIMA could perform better.

Similarly, among Chaos-based hybrids (Chaos+CART, Chaos+RF, Chaos+CNN), the hybrid, Chaos+RF, could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better than Chaos+CART.

Figure 7 depicts predictions of the test set of Shanghai Composite Index. The predictions are obtained from CNN, Chaos+CNN, and Chaos+CNN+PR. From the figure, it can be observed that the predictions obtained using Chaos+CNN+PR are very much closer to actual values. It is also worth noting that the predictions obtained using CNN are better than that of Chaos+CNN.

### 5.7 Crude oil price

The Crude Oil Price test set results of prediction approaches are presented in Table 13. The table reveals that the proposed hybrid, Chaos+CNN+PR, outperformed all other approaches in terms of MSE, MAPE, Dstat and Theil’s $U$. The performance measures MSE, MAPE and Theil’s $U$ are very much closer to 0 indicate that predictions are very much closer to actual values. Dstat value 100 indicates that directional change is fully captured by the proposed hybrid.

Among the standard prediction approaches (ARIMA (2,1,0), Prophet, CNN, CART, RF), RF followed by CNN could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better. In this context, ARIMA could perform better.

Similarly, among Chaos-based hybrids (Chaos+CART, Chaos+RF, Chaos+CNN), the hybrid, Chaos+RF, could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better than Chaos+CNN.

Figure 8 depicts predictions of the test set of Crude Oil Price. The predictions are obtained from CNN, Chaos+CNN, and Chaos+CNN+PR. From the figure, it can be observed that the predictions obtained using Chaos+CNN+PR are very much closer to actual values. It is also worth noting that the predictions obtained using CNN are better than that of Chaos+CNN.

### 5.8 Gold price

The Gold Price test set results of prediction approaches are presented in Table 14. The table reveals that the proposed hybrid, Chaos+CNN+PR, outperformed all other approaches in terms of MSE, MAPE, Dstat and Theil’s $U$. The performance measures MSE, MAPE and Theil’s $U$ are very much closer to 0 indicate that predictions are very much closer to actual values. Dstat value 100 indicates that directional change is fully captured by the proposed hybrid.

Among the standard prediction approaches (ARIMA (2,1,1), Prophet, CNN, CART, RF), CNN followed by RF could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better. In this context, CART could perform better.

Similarly, among Chaos-based hybrids (Chaos+CART, Chaos+RF, Chaos+CNN), the hybrid, Chaos+RF, could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better than Chaos+CNN.

Figure 9 depicts predictions of the test set of Gold Price. The predictions are obtained from CNN, Chaos+CNN, and

|  | Forecasting model | MSE       | MAPE    | DStat       | Theil $U$  |
|---|------------------|-----------|---------|-------------|------------|
| ARIMA | 668777.360368 | 19.187506 | 53.840924 | 0.027243    |
| Prophet | 2265404.326675 | 87.251734 | 51.597552 | 0.172626    |
| CNN | 2956.134975 | 1.181762 | 48.130523 | 0.000146    |
| CART | 8076.550985 | 1.950799 | 53.840924 | 0.000397    |
| RF | 3623.417136 | 1.256007 | 49.490142 | 0.000178    |
| Chaos+CART | 7433.170556 | 1.879328 | 55.268524 | 0.000365    |
| Chaos+RF | 3598.312050 | 1.246783 | 49.558123 | 0.000176    |
| Chaos+CNN | 35844.767975 | 3.705842 | 49.694085 | 0.001777    |
| Chaos+CNN+PR | 0.047778 | 0.006898 | 100.0 | 0.000009 |
Chaos + CNN. From the figure, it can be observed that the predictions obtained using Chaos + CNN + PR are very much closer to actual values. It is also worth noting that the predictions obtained using CNN are better than that of Chaos + CNN.

5.9 Soya beans price (USD)

The Gold Price test set results of prediction approaches are presented in Table 15. The table reveals that the proposed hybrid, Chaos + CNN + PR, outperformed all other approaches in terms of MSE, MAPE, Dstat and Theil’s U. The performance measures MSE, MAPE and Theil’s U are very much closer to 0 indicate that predictions are very much closer to actual values. Dstat value 100 indicates that directional change is fully captured by the proposed hybrid.

Among the standard prediction approaches (ARIMA (0,1,0), Prophet, CNN, CART, RF), CNN followed by RF could yield better predictions in terms of MSE, MAPE and Theil’s U. However, it could not capture the

---

Table 13 Test set results of crude oil price

| Forecasting model | MSE        | MAPE       | DStat     | Theil U     |
|-------------------|------------|------------|-----------|-------------|
| ARIMA             | 122.1701375| 15.62913992| 52.75840203| 0.022626621 |
| Prophet           | 2079.515013| 43.66922359| 50.15852885| 0.181741948 |
| CNN               | 3.917877221| 2.501259507| 49.58782498| 0.00072788  |
| CART              | 5.981146247| 3.441771962| 51.68040583| 0.001107432 |
| RF                | 3.906917266| 2.439084782| 48.82688649| 0.000722327 |
| Chaos+CART        | 5.859357558| 3.400616935| 51.61699429| 0.001084689 |
| Chaos+RF          | 3.898882222| 2.432118422| 49.39759036| 0.000720849 |
| Chaos+CNN         | 10.43330561| 6.553971389| 52.69499049| 0.00194532  |
| Chaos+CNN+PR      | 1.55E – 07 | 0.000819526| 100       | 2.87E – 11  |
direction change better. In this context, CART could perform better.

Similarly, among Chaos-based hybrids (Chaos+CART, Chaos+RF, Chaos+CNN), the hybrid, Chaos+RF, could yield better predictions in terms of MSE, MAPE and Theil’s $U$. However, it could not capture the direction change better than Chaos+CART.

Figure 10 depicts predictions of the test set of Gold Price. The predictions are obtained from CNN, Chaos+CNN, and Chaos+CNN+PR. From the figure, it can be observed that the predictions obtained using Chaos+CNN+PR are very much closer to actual values. It is also worth noting that the predictions obtained using CNN are better than that of Chaos+CNN.

Finally, the Diebold and Mariano test [8] is used to officially test the statistical difference between Chaos+CNN+PR and other forecast models on average. The test of statistical significance accepts the predictions obtained from two approaches as inputs. Table 16 shows

| Table 14 Test set results of gold price |
|----------------------------------------|
| Forecasting model | MSE         | MAPE        | DStat       | Theil $U$   |
| ARIMA             | 27802.054680 | 9.048922    | 51.708860   | 0.006927    |
| Prophet           | 67822.883460 | 15.076782   | 50.569620   | 0.015280    |
| CNN               | 347.445571   | 0.966208    | 46.582278   | 8.670003e-05|
| CART              | 2650.180641  | 1.867481    | 58.037974   | 0.000671    |
| RF                | 1389.370580  | 1.279991    | 51.455696   | 0.000351    |
| Chaos+CART        | 2660.907862  | 1.888391    | 57.341772   | 0.000674    |
| Chaos+RF          | 1530.644763  | 1.329603    | 50.506329   | 0.000387    |
| Chaos+CNN         | 4267.047461  | 2.839465    | 51.202531   | 0.001080    |
| Chaos+CNN+PR      | 0.003165     | 0.004098    | 100.0       | 7.944620e-10|
the absolute values of the Diebold-Mariano test statistic for each of the nine datasets. If the absolute value of the test statistic is less than or equal to 1.96, the corresponding model is equivalent to Chaos+CNN+PR. The table clearly shows that Chaos+CNN+PR outperforms every model for every dataset as all of the absolute values of test statistic are greater than 1.96.

6 Conclusion

A novel hybrid model, Chaos+CNN+PR, is presented in this paper to resolve to predict financial time series. The financial time series in this Hybrid is first checked for chaos. Later on, Chaos Theory can model chaos in the time series. Input to CNN is used to create initial predictions for the model time series. The CNN predictions error series is input to PR to get error predictions. The error predictions and initial CNN predictions are added to produce final

Table 15 Test set results of soya beans price

| Forecasting model | MSE        | MAPE      | DStat    | Theil U      |
|-------------------|------------|-----------|----------|--------------|
| ARIMA             | 36277.43250| 15.161464 | 50.775915| 0.016791     |
| Prophet           | 317374.435389| 36.564604 | 50.900062| 0.099628     |
| CNN               | 124.167554 | 0.817279  | 51.024208| 6.779597e – 05|
| CART              | 492.448338 | 1.732297  | 55.307262| 0.000026     |
| RF                | 186.908517 | 1.072228  | 51.707014| 0.000010     |
| Chaos+CART        | 483.229258 | 1.730958  | 55.493482| 0.0000263    |
| Chaos+RF          | 186.237605 | 1.065448  | 52.203600| 0.0000101    |
| Chaos+CNN         | 1627.197407| 3.230644  | 51.893234| 0.000890     |
| Chaos+CNN+PR      | 1.336608e – 05| 0.000306  | 100.0    | 7.288284e – 12|
predictions. Three kinds of financial time series, such as foreign exchange, commodity, and stock market indices, are used to test the proposed Hybrid’s effectiveness. The proposed hybrid, in terms of MSE, MAPE, Dstat, and Theil’s $U$, outperformed ARIMA, Prophet, CNN, CART, RF, Chaos+CNN, Chaos+CART, and Chaos+RF. It is also possible to extend the proposed Hybrid to various financial and non-financial time series. The regression problem solved here can also be converted into a classification problem. In this context, the approaches proposed by [40–42] are very much helpful.
Declarations

Conflict of interest The authors declare that they have no conflict of interest with any author, or organization.

References

1. Akaike H (1974) A new look at the statistical model identification. IEEE Trans Autom Control 19(6):716–723
2. Bahrammirzaee A (2010) A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems. Neural Comput Appl 19(8):1165–1195
3. Bao W, Yue J, Rao Y (2017) A deep learning framework for financial time series using stacked autoencoders and long-short term memory. PloS One 12(7):e0180944
4. Bates JM, Granger CW (1969) The combination of forecasts. J Oper Res Soc 20(4):451–468
5. Cao L (1997) Practical method for determining the minimum embedding dimension of a scalar time series. Phys D Nonlinear Phenom 110(1–2):43–50
6. Cavalcante RC, Brasilheiro RC, Souza VL, Nobrega JP, Oliveira AL (2016) Computational intelligence and financial markets: a survey and future directions. Expert Syst Appl 55:194–211
7. Dhanya C, Kumar DN (2010) Nonlinear ensemble prediction of chaotic daily rainfall. Adv Water Resour 33(3):327–347
8. Diebold F, Mariano R (1995) Comparing predictive accuracy. J Bus Econ Stat 13:253–263
9. Durairaj M, Mohan BK (2019) A review of two decades of deep learning hybrids for financial time series prediction. Int J Emerg Technol 10(3):324–331
10. Heaton J, Polson NG, Witte JH (2016) Deep learning in finance. arXiv preprint arXiv:160206561
11. Hellström T, Holmström K (1998) Predicting the stock market. Technol 10(3):324–331
12. Huang W, Lai KK, Nakamori Y, Wang S (2004) Forecasting foreign exchange rates with artificial neural networks: a review. Int J Inf Technol Decis Mak 3(01):145–165
13. Huang W, Lai KK, Nakamori Y, Wang S (2004) Forecasting foreign exchange rates with artificial neural networks: a review. Int J Inf Technol Decis Mak 3(01):145–165
14. LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, Hubbard W, Jackel LD (1989) Backpropagation applied to handwritten zip code recognition. Neural Comput 1(4):541–551
15. LeCun Y, Bengio Y et al (1995) Convolutional networks for image recognition. IEEE Trans Comput C-36(11):1233–1242
16. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444
17. Li Y, Ma W (2010) Applications of artificial neural networks in financial economics: a survey. In: 2010 International symposium on computational intelligence and design, IEEE, vol 1, pp 211–214
18. Livieris IE, Pintelas E, Pintelas P (2020) A CNN-LSTM model for gold price time-series forecasting. Neural Comput Appl 32(23):17351–17360
19. Magdon-Ismail M, Nicholson A, Abu-Mostafa YS (1998) Financial markets: very noisy information processing. Proc IEEE 86(11):2184–2195
20. Makridakis S, Hibon M (1995) Evaluating accuracy (or error) measures. Insead 110:1–41
21. Makridakis S, Hibon M (1997) ARMA models and the Box–Jenkins methodology. J Forecast 16(3):147–163
22. Mochón A, Quintana D, Sáez Y, Isasi P (2008) Soft computing techniques applied to finance. Appl Intell 29(2):111–115
23. Packard NH, Crutchfield JP, Farmer JD, Shaw RS (1980) Geometry from a time series. Phys Rev Lett 45(9):712
24. Pavlidis N, Tasoulis D, Vrahatis MN (2003) Financial forecasting through unsupervised clustering and evolutionary trained neural networks. In: The 2003 congress on evolutionary computation, 2003. CEC’03., IEEE, vol 4, pp 2314–2321
25. Pfeiffer M, Hohmann A (2012) Applications of neural networks in training science. Hum Mov Sci 31(2):344–359
26. Poincaré H (1890) Sur le problème des trois corps et les équations de la dynamique. Acta Math 13(1):A3–A270
27. Pradeepkumar D, Ravi V (2014) Forex rate prediction using chaos, neural network and particle swarm optimization. In: International conference in swarm intelligence, Springer, pp 363–375
28. Pradeepkumar D, Ravi V (2016) Forex rate prediction using chaos and quantile regression random forest. In: 2016 3rd international conference on recent advances in information technology (RAIT), IEEE, pp 517–522
29. Pradeepkumar D, Ravi V (2017) Forex rate prediction: a hybrid approach using chaos theory and multivariate adaptive regression splines. In: Proceedings of the 5th international conference on frontiers in intelligent computing: theory and applications, Springer, pp 219–227
30. Pradeepkumar D, Ravi V (2018) Soft computing hybrids for forex rate prediction: a comprehensive review. Comput Oper Res 99:262–284
31. Ravi V, Pradeepkumar D, Deb K (2017) Financial time series prediction using hybrids of chaos theory, multi-layer perceptron and multi-objective evolutionary algorithms. Swarm Evolut Comput 36:136–149
32. Reid DJ (1968) Combining three estimates of gross domestic product. Economica 35(140):431–444
33. Selvin S, Vinayakumar R, Gopalakrishnan E, Menon VK, Soman K (2017) Stock price prediction using LSTM, RNN and CNN-sliding window model. In: 2017 international conference on advances in computing, communications and informatics (iccci), IEEE, pp 1643–1647
34. Sezer OB, Gudelek MU, Ozbayoglu AM (2020) Financial time series forecasting with deep learning: a systematic literature review: 2005–2019. Appl Soft Comput 80:106–181
35. Takens F (1981) Detecting strange attractors in turbulence. In: Rand D, Young LS (eds) Dynamical systems and turbulence, Warwick 1980. Lecture Notes in Mathematics, vol 898. Springer, Berlin, Heidelberg. https://doi.org/10.1007/BFb0091924
36. Theil H (1966) Applied economic forecasting. North-Holland Pub. Co., Amsterdam
37. Vidal A, Kristjanpoller W (2020) Gold volatility prediction using a CNN-LSTM approach. Expert Syst Appl 157:113481
38. Williams G (1997) Chaos theory tamed. CRC Press, Boca Raton
39. Yao J, Tan CL (2000) A case study on using neural networks to perform technical forecasting of forex. Neurocomputing 34(1–4):79–98
40. Yu X, Yang J, Xie Z (2014) Training SVMs on a bound vectors set based on fisher projection. Front Comput Sci 8(5):793–806
41. Yu X, Chu Y, Jiang F, Guo Y, Gong D (2018) SVMs classification based two-side cross domain collaborative filtering by inferring intrinsic user and item features. Knowl Based Syst 141:80–91
42. Yu X, Jiang F, Du J, Gong D (2019) A cross-domain collaborative filtering algorithm with expanding user and item features via the latent factor space of auxiliary domains. Pattern Recognit 94:96–109

43. Zhang K, Zhong G, Dong J, Wang S, Wang Y (2019) Stock market prediction based on generative adversarial network. Procedia Comput Sci 147:400–406