Research Article

Time series forecasting of agricultural product prices based on recurrent neural networks and its evaluation method

Koichi Kurumatani[1] 🄳

Abstract

We propose a time series forecasting method for the future prices of agricultural products and present the criteria by which forecasted future time series are evaluated in the context of statistical characteristics. Time series forecasting of agricultural products has the basic importance in maintaining the sustainability of agricultural production. The prices of agricultural products show seasonality in their time series, and conventional methods such as the auto-regressive integrated moving average (ARIMA or the Box Jenkins method) have tried to exploit this feature for forecasting. We expect that recurrent neural networks, representing the latest machine learning technology, can forecast future time series better than conventional methods. The measures used in evaluating the forecasted results are also of importance. In literature, the accuracy determined by the error rate at a specific time point in the future, is widely used for evaluation. We predict that, in addition to the error rate, the criterion for conservation of the statistical characteristics of the probability distribution function from the original past time series to the future time series in the forecasted future is also important. This is because some time series have a non-Gaussian probability distribution (such as the Lévy stable distribution) as a characteristic of the target system; for example, market prices on typical days change slightly, however on certain occasions, change dramatically. We implemented two methods for time series forecasting based on recurrent neutral network (RNN), one of which is called time-alignment of time point forecast (TATP), and another one is called direct future time series forecast (DFTS). They were evaluated using the two aforementioned criteria consisting of the accuracy and the conservation of the statistical characteristics of the probability distribution function. We found that after intensive training, TATP of LTSM shows superior performance in not only accuracy, but also the conservation compared to TATP of other RNNs. In DFTS, DFTS of LSTM cannot show the best performance in accuracy in RMS sense, but it shows superior performance in other criteria. The results suggest that the selection of forecasting methods depends on the evaluation criteria and that combinations of forecasting methods is useful based on the application. The advantage of our method is that the required length of time series for training is enough short, namely, we can forecast the whole cycle of future time series after training with even less than the half of the cycle, and it can be applied to the field where enough numbers of continuous data are not available.

✉ Koichi Kurumatani, k.kurumatani@aist.go.jp; k.kurumatani@gmail.com | ¹Human Augmentation Research Center (HARC), National Institute of Advanced Industrial Science and Technology (AIST), Aomi 2-3-26, Koto-ku, Tokyo 135-0064, Japan.

SN Applied Sciences
A SPRINGER NATURE journal

1 Introduction

Time series forecasting of agricultural products play a major role in the sustainability of agricultural production. Providing price forecasting information would help in decision making for managing agricultural supplies and helping to improve purchasing behaviors of consumers. In addition to the seasonality in production side caused by year-round cycle climate change, the prices are affected by the users' preferences to products and suppliers' trading strategy and behavior. These human factors do not always have seasonality but they could be represented as relations between events in past and the current status.

Currently, forecasts are being carried out using the conventional Box-Jenkins method [1]. In this paper, we propose forecasting methods based on the recurrent neural network (RNN) [2] for forecasting future commodity prices. There is seasonality in commodity markets that bring about cyclic value changes depending on the season. Conventional approaches in time series forecasting by auto-regressive integrated moving average (ARIMA) have tried to exploit this seasonality among other dependencies for neighboring points observed in the original past sequences by linear modeling [1].

The latest machine learning technologies, including the RNN, are expected to extract hidden relations in target systems, and demonstrate a higher performance than the conventional methods because (1) RNN enables non-linear modeling of the target sequence, (2) the gate mechanism in long short-term memory (LSTM) [3] and gated recurrent unit (GRU) [4] families provide a way of modeling the direct relations between a specific past time point and the current time point.

We implemented two methods for time series forecasting based on RNN, one of which is called time-alignment of time point forecast(TATP), and the other one is called direct future time series forecast (DFTS).

The criteria for evaluating the forecasted results are also important. We used three types of evaluation criteria: two types of accuracy by error measurements of root mean square (RMS) and mean absolute (MA), and the conservation performance of probability distribution function (PDF). We found it insufficient to only use the forecast error on the time point to evaluate the performance of the forecasting method, because time series of large complex systems, such as social systems, show macro-behaviors that cannot be characterized by time point errors.

This study is organized as follows: after reviewing previous literature in Sect. 2, Sect. 3 shows the data set used in this study which includes the agricultural product price history. After describing our time series forecasting method in Sect. 4, the forecasted results are shown in Sect. 5. The evaluation accuracy results are given in Sect. 6 and the conservation performance of statistical characteristics are provided in Sect. 7. After comparing proposed methods in Sect. 8 and discussing the findings of the study in Sect. 9, we conclude the study in Sect. 10.

2 Related work

One of the most important conventional methods for future time series forecasting methods is the Box-Jenkins method [1], which is based on a linear combination of weighted past values and the explicit introduction of seasonality describing repeated value changes in seasonal cycles.

From previous literature on combining machine learning and Box-Jenkins method, improvements on the Box-Jenkins and its related methods (autoregressive [AR], autoregressive integrated moving average [ARIMA], SARIMA) have been proposed through addition of artificial neural network (ANN) to capture the non-linearity in the target system [5–8]. The most used method in this direction is sequential hybrid approach that can be refined in the sense of accuracy by introducing weighted sequential hybrid methodology [9]. It is important to clarify the structure of a forecasting method as the combination of a conventional linear method and an ANN, which improves the easiness of training of ANN. There is a possibility of improving the accuracy by introducing gate mechanism of latest non-linear neural networks.

About the improvement of forecasting accuracy by neural networks, a method for controlling the influences from past sequences, by using two convolutional neural networks (CNN) for AR and the so-called offset for influence control, has been proposed and has displayed high accuracy [10]. Improved versions of LSTM such as Phased LSTM [11], Augmented LSTM [12], a temporal convolution-based algorithm called WaveNet [13], and a new reinforcement learning-based prediction algorithm [14], have also been proposed. DeepAR [15] is a framework to handle forecasting future values as probability distribution. The basic forecasting method is based on linear autoregressive models. LSTNet [16] is a hybrid network consisting (1) convolutional layer, (2) recurrent layer, and (3) fully-connected autoregressive (AR) output integration, the advantage of which is to clarify the role of each part and to make the training easier. The representation power is, however, equal to temporal convolution [13] and autoregressive model. Although these methods have displayed high accuracy in forecasting values at a specific point in the future, the statistical characteristics of the forecasted future time series have not been evaluated. The aim of our

research is to establish forecasting methods that satisfy both high accuracy and natural statistical characteristics that is shown in the original time series.

In order to overcome the shortcomings of conventional autoregressive methods about the limitation of numbers to be simultaneously forecasted, matrix factorization-based method [17], unsupervised scalable representation learning [18], DeepGLO that combines a global matrix factorization and a local temporal network [19] have been proposed. These methods explore the huge space of training data consisting of higher dimensions of variables and temporal duration. About the problem of forecasting agricultural product price at a nation level, we have to carry out the forecasting based on limited numbers of sampled data that has missing part of duration and/or product item, where these methods cannot be directly applied.

There are a couple of approaches to agricultural product price by machine learning, Introducing simple RNN (they call it "Time-Delay Neural Network") in agricultural product price forecasting [20] sometimes improves the accuracy. There are approaches to introduce (1) Fuzzy and SVM [21] and (2) Naïve Baysian Algorithm [22], where the evaluation methods are limited and the advantages against conventional methods are unclear.

The aim of this research is to establish a method with the following characteristics.

(1) It can enable the selection of basic forecasting mechanism such as simple RNN, LSTM, GRU, and so on, according to the specified criteria such as time-point accuracy, global accuracy, and statistical characteristics of generated future time series.

(2) For the selection, it clarifies the effectiveness and limitation of controlling the influence between a specific past time point and the present using the gate (such as forget, peephole) mechanism in LSTM and GRU.

(3) The algorithm we propose must fit the limitation of relatively small numbers of training data that has missing part of temporal duration and product.

In literature, there is an approach to compare the accuracies of multi-step ahead forecasting by multiple-output and single-output neural networks [23]. Our basic idea of selecting suitable forecasters has the same essence of their approach, although our approach includes latest improvements of neural networks such as gate algorithms and new evaluation methods.

3 Data of agricultural product price

We have considered the price of agricultural products as our example because they are indispensable commodities for human life, and also because their price is affected by several aspects including past price sequence, sales, supply amount, consumer preference, weather, and so on (in Fig. 1). We describe the first step in the forecast mechanism, which is forecasting future price values by the past price history only.

The restriction on forecasting agricultural product price at a nation level is the limited amount of data, in contrast with international agricultural product markets. We need to develop algorithms that work with less amount of data with controlling the quality of forecasted results from multiple-points of view.

We used the Japanese government price movement data of agricultural products. The price data are collected by the Ministry of Agriculture, Forestry, and Fisheries, which are open to the public [24]. We focused on the representative types of agricultural products, especially the types of vegetables that are indispensable and popular in daily life. The coverage and the system of collecting the government data are as follows: the price of the target vegetable is measured at a total 470 retail stores, with 10 stores in each of the 47 prefectures in Japan (the earlier data was obtained from 460 stores in 46 prefectures). The frequency of measurement is once per week. These 470 price values measured in a specific week are arithmetically averaged and the obtained price information is published on a website. The price data starting from the
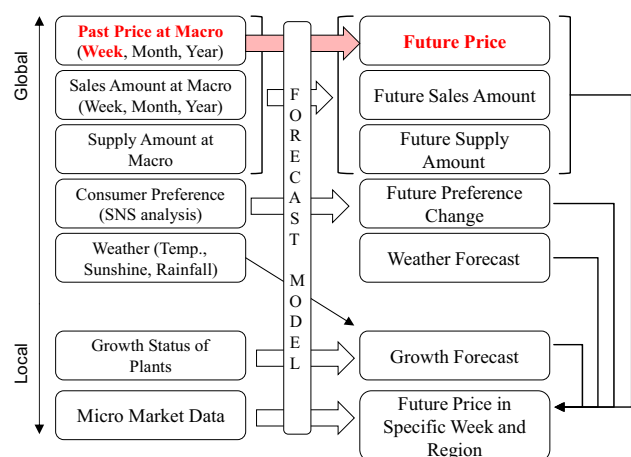

Fig. 1 Whole forecasting framework. Upon receiving the past price time series, sales and supply amount, consumer preference, weather, and so on, the forecast model generates the future values. In this paper, we only worked with market price data shown in red color

week beginning April 12, 2010 onwards is available. There is missing data for some periods due to the disruption by the Great East Japan Earthquake (on March 11, 2011) in Tohoku area and policy changes.

The selection of target agricultural products varies depending on the period of measurement. The most representative products are chosen and measured for specified periods. We selected three vegetables in the data—cabbage, tomato, and lettuce—for this experiment, because (1) these vegetables are popular in daily life, (2) their sale volumes is high, and (3) their data is readily available (covering a maximum of 270 weeks in total) in the whole surveillance data. For the training of learning mechanism, we used price data from the week beginning April 12, 2010 to the week beginning February 5, 2018. The forecasting and evaluation of the results for unknown test data were carried out for the period between February 12, 2018 and February 11, 2019.

(a) Time-Alignment of Time Point Forecast (TATP)
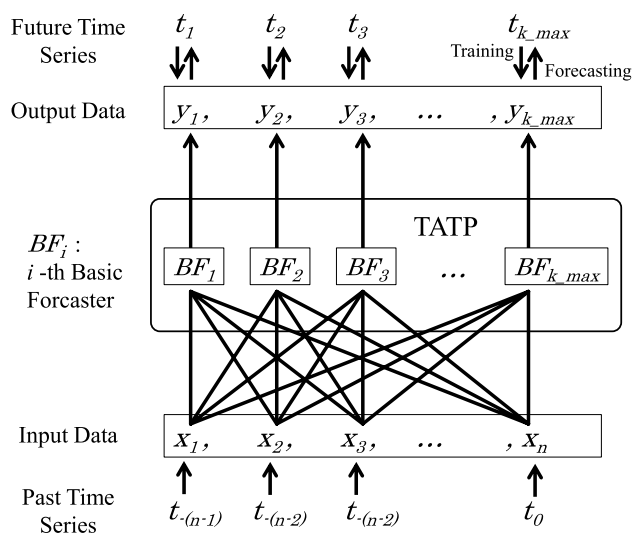
4 Method of time series forecasting

In this paper, we only worked with discrete time series. The time series forecasting can be expressed mathematically as a mapping (function) from a past sequence that is an ordered set of vector values $\{x_i\}$ consisting of past values to a future sequence that is also an ordered set of vector values $\{y_j\}$, where a sequence is a part of the whole time series.

There are two ways to generate the future sequence $\{y_j\}$ from a past one $\{x_i\}$ by RNN depending on the type of combining generating functions. The first one is by combining time point forecast results into a future sequence. In this method, first, a forecast mechanism generates a value independently for each point in the future, and next, the generated values are aligned in a sequence for the future sequence. We call this method TATP. The second way is by forecasting the future sequence as a whole by setting the future sequence itself as the output data for learning. We call this method DFTS. The structure of TATP and DFTS are shown in Fig. 2.
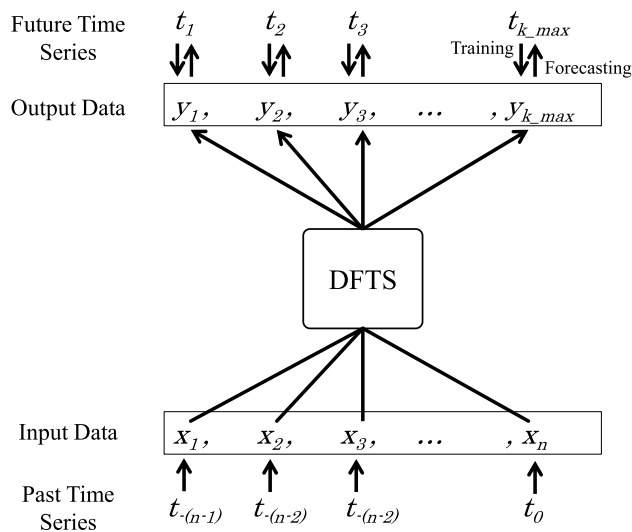
4.1 Basic forecaster of TATP

The basic forecaster of TATP means an ANN that is trained by the past time series and provides the value of a specific time point in future. A basic forecaster receives an n-step past sequence (time series of past values) and returns the value at 'k-step after'.

The basic idea is as follows: we prepare basic forecaster $p(n, k)$ that receives an n-step length of past price sequence and returns the price after k-steps in the future. A basic forecaster can be applied for any n-step length part

(b) Direct Future Time Series Forecast (DFTS)

Fig. 2 Time series forecasting frameworks of TATP and DFTS

of a past sequence s_j^p. The predicted price value at 'k-step after' the past sequence s_j^p by using a basic forecaster $p_i(n, k)$ is denoted as $p_i(n, k)(s_j^p)$.

Each $p_i(n, k)$ is an ANN that is trained to forecast 'k-step after' from an n-step past sequence. We used LSTM, GRU, and simple recurrent neural network (SRNN) to generate the networks. The duration of prediction of k varied from 1 to k_max. Because the past data was available on weekly basis, we set the duration of the forecasting time step to be 1 week.

In this study, we fixed the duration of the past sequence n to be 20. This means that basic forecasters were trained

by the data of the previous 20 weeks. The first reason why we chose the duration of 20 is for evaluation purposes. Agricultural product prices have seasonal cycles meaning that the price changes according to the season, and the longest seasonal cycle is 1 year. By setting the training period to be shorter than 1 year, we could evaluate the ability of forecasting mechanism in the case where the data is available for less duration than the longest cycle. In our case, the duration 20 weeks is about one-third of a year. The second reason is for data availability. Because there are missing periods and missing values in the original data set and it is unsuitable to insert interpolated values, we need to select a relatively shorter period for data preparation.

Using this method, we prepared 175 training data for RNNs from the past time series. The duration of future forecasting, k_max was fixed at 53. This value was to obtain future forecasting results of about 1 year.

Forecasting of 'k-step after' week was carried out with RNNs (LSTM, GRU, and simple RNN), each of which was trained by a set of past sequence and the price value at 'k-week after' where k is an integer value in [1, 53]. RNN with one middle layer of 50 units was trained with stochastically chosen 90% training data (153 samples) from the past price sequence, and the rest 10% (17 samples) was used for validation. The total number of original sequences was 312 weeks, and continuous parts were chosen for training and validation data. The optimizer was Adam [25].

Back-propagation learning converged to stability for approximately 200 to 300 epochs for all types of RNNs. We trained for up to 10,000 epochs and found that GRU overfitted after 200 epochs and SRNN overfitted after 300 epochs. LSTM showed improvement during the training up to 10,000 epochs and showed the highest performance in the latter part of the epochs.

LSTM with a gate mechanism connecting a certain time point in the past to the present succeeded in finding sharp convex (peaks) that correspond to a sudden rise or fall in changes in the values in the time series, even for the long-term prediction 15 weeks after, reported in [26]. However, GRU cannot capture such sharp changes embedded in the original time series. In this previous work [26], TATP-based forecasting and its results within less than 1-year cycle was described. In this paper, both TATP and DFTS within just 1-year cycle and their results are studied.

Accuracy of validation data was less than 0.01 for all basic forecasters, which means that they showed high performance for 10% validation data in the original past sequence.

After the training, 30 best results with lowest validation loss were selected among the trained ones, that is, i_max was 30. As a result, we obtained a set of 1590 basic forecasters $\{p_i(n, k)|\ i$ in [1, 30], $n = 20$, k in [1, 53]$\}$ for each RNN, that is, LSTM, GRU, and SRNN.

4.2 TATP: construction of time series forecaster

We construct a time series forecaster by the time alignment of basic forecasters as follows. For each k, after obtaining enough number of i_max sets of basic forecasters $p_i(n, k)$ (i in [1, 2, …, i_max]), we sort the basic forecasters in the order of their performance (in terms of root mean square error [RMSE]). After collecting the best forecasters from each k between 1 and k_max, we construct the sequence of the forecasters in the order of k:

$$S^f(1) = [p(n, 1), p(n, 2), \ldots, p(n, k\_max)]. \tag{1}$$

Similarly, the sequences constructed by the second and third best performer is denoted by $S^f(2)$, $S^f(3)$, respectively. In this way, we obtain $S^f(i)$ ($i = [1, 2, …, i\_max]$). The forcasted sequence values in the future from the past sequence s_j^p is denoted by $S^f(i)(s_j^p)$.

4.3 DFTS

The second forecasting method is called DFTS, in which the time series in the future itself is directly forecasted by an ANN that has been trained by the set of past time series sequences as input of RNN and the future sequences as the output. The forecasted future sequence is denoted as

$$S^f(1) = [v_1, v_2, \ldots, v_n], \tag{2}$$

where $S^f(1)$ is the best forecast performer with the best performance (in terms of RMSE), and $S^f(1)$, $S^f(2)$, …, $S^f(i)$ for the second one, and so on.

The best TATP and DFTS forecasters are not identical. In TATP, each basic forecaster is trained to achieve the highest performance for a specific point in the future, that is, the performance with the least error for the specific time point. In contrast, the DFTS forecaster is trained to simultaneously decrease the error for all points in the future sequence. Therefore, forecasters obtained by TATP and DFTS from the same data set behave differently.

The details of learning settings of DFTS are the same as those for TATP. For DFTS of all types of RNNs (LSTM, GRU, and SRNN), the learning converges approximately 1000 epochs. Loss of validation is 0.000902481, 0.001141842, and 0.000888557 for LSTM, GRU, and SRNN, respectively.
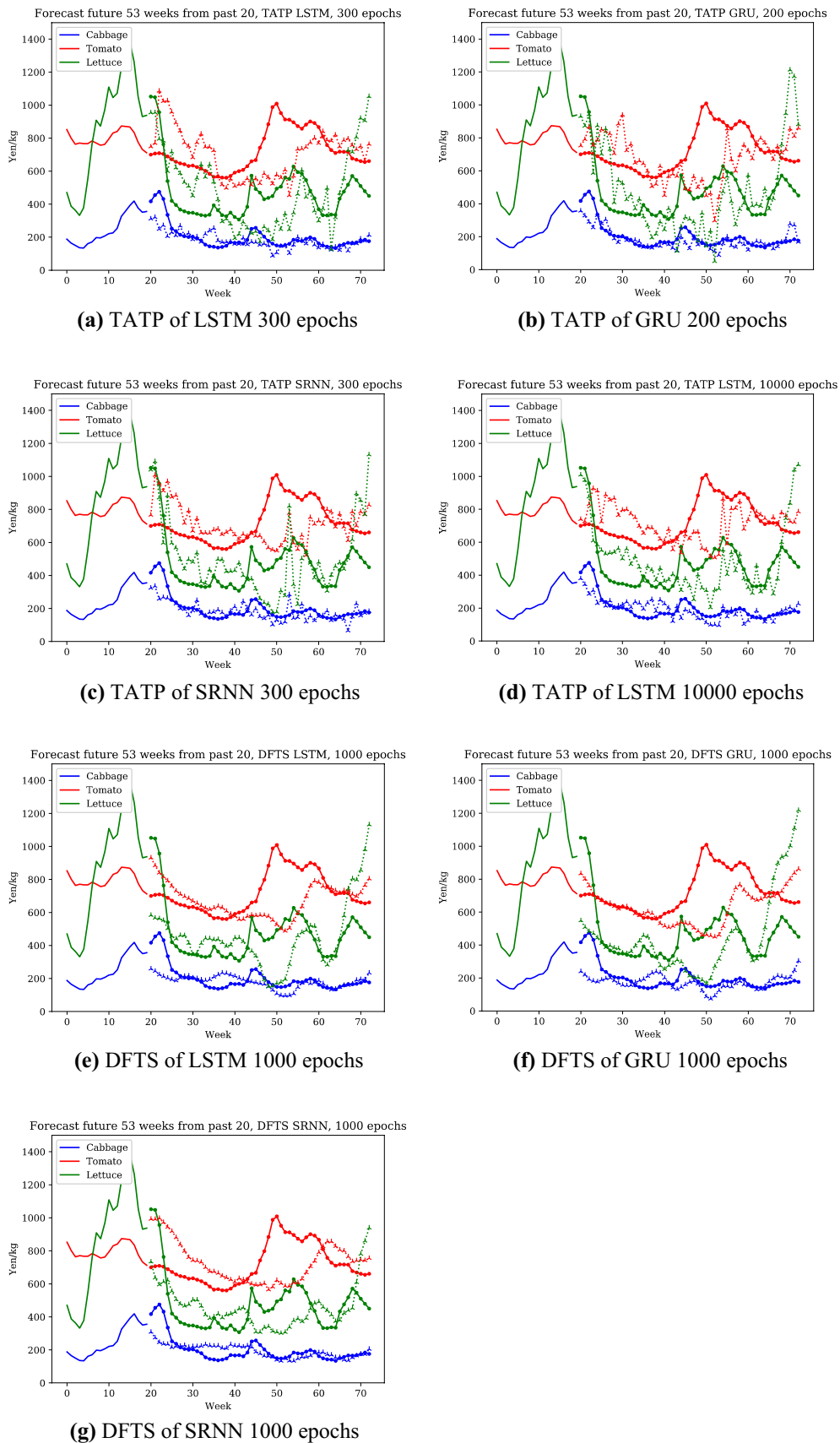
(a) TATP of LSTM 300 epochs

(b) TATP of GRU 200 epochs

(c) TATP of SRNN 300 epochs

(d) TATP of LSTM 10000 epochs

(e) DFTS of LSTM 1000 epochs

(f) DFTS of GRU 1000 epochs

(g) DFTS of SRNN 1000 epochs

Fig. 3 Forecasted prices 53 weeks into the future

5 Forecasted future time series by TATP, DFTS

Examples of forecasted future time series by TATP and DFTS are shown in Fig. 3. All RNNs were trained by the same time series before February 12, 2018. In the figures, the first 20-week data represents the ground truth data (real values) before February 12, 2018. A solid line represents ground truth data (real values) for 53 weeks from February 12, 2018 to February 11, 2019. A broken line represents forecasted values for the same period.

Regarding the TATP of GRU and SRNN, learning converged at the early stages. The results by forecasters after 200 (GRU) and 300 (SRNN) epochs training are shown in Fig. 3b, c, respectively. For TATP of LSTM, learning progresses after 300 epochs, and it seems to converge at about 10,000 epochs. The results by forecasters after 300 and 10,000 epochs training are shown in Fig. 3a, d, respectively.

TATP of all types of RNNs are able to capture long-term tendency (seasonality) of value changes in 53 weeks. For week 1 up to week 15, the separation between the real and forecasted values was substantially larger than other weeks for lettuce and tomato. All the types of RNNs, however, showed a similar tendency that forecasted values were higher than the real values. Therefore, there is a possibility that the real price was lower than usual in the previous years.

When LSTMs were compared for within 300 (LSTM-300) and within 10,000 (LSTM-10k) epochs, LSTM-10k showed higher accuracy at many points in the sequence, but not at all of them.

For DFTS, all three types of LSTM, GRU, and SRNN converged within 1000 epochs. We can qualitatively establish the tendency of DFTS results in comparison with that of TATP results. The first tendency is that the forecasted results of DFTS are smoother than those of TATP meaning that the value differences between two adjacent time points are smaller compared with those of TATP. The reason is that since in TATP, the value at a future time point is learned independently from that at other future time points, there are no direct relations between the values in the forecasted future time series. In contrast, in DFTS, all the values at future time points are learned simultaneously and the direction of back-propagation is determined so as to simultaneously decrease the loss of all values equally. This simultaneousness seems to put relations between adjacent or neighboring points closer. These characteristics of smoothness will be discussed in the context of PDF in Sect. 7.

The second tendency in the forecasted time series is the similarity in the capability of all algorithms to capture the macro-behavior of time series, meaning that all the algorithms can capture the value changes in long term cycles. For instance, although the real-observed price value of tomato (red line) is highest at approximately week 50 in the graph, the forecasted values by all algorithms were much lower than those observed. This suggests that all the algorithms succeeded in capturing the long-term cyclic behavior in the past time series and forecasted the future precisely only when the tendency is still effective in this test case. In other words, the real tomato price in this test case might be higher than the usual prices in the past history due to some special reasons for this year.

The forecasting system was implemented using Keras, TensorFlow, and Python on Windows 7 and Linux.

6 Evaluation in accuracy

Our time series forecasting method, TATP and DFTS on LSTM, GRU, and SRNN can generate future time series. However, the problem is how to evaluate the forecasted results in the context of comparison with real values.

We need an accurate and effective method to evaluate these forecasted results when compared with real values. In this section, we evaluate the forecasted future time series based on accuracy.
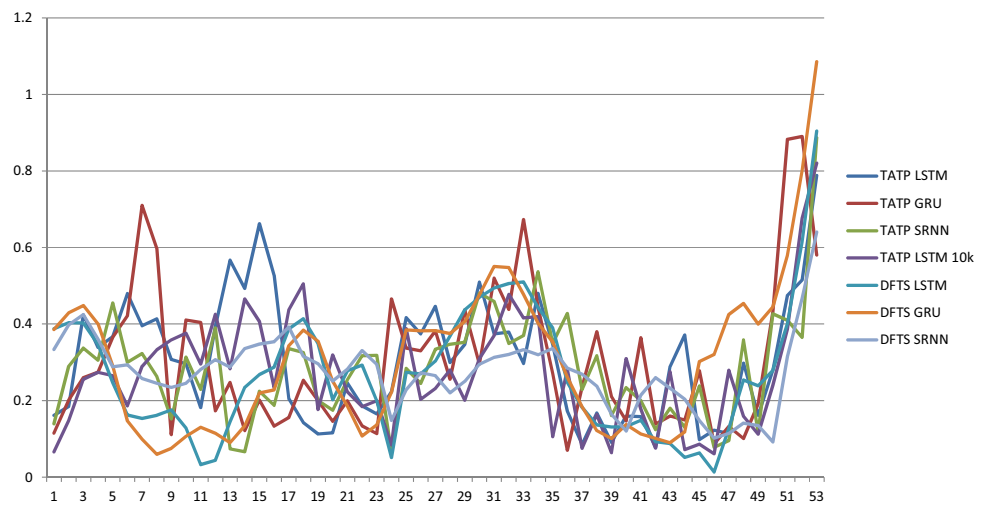
6.1 Time point accuracy

The accuracy for time series should be measured in two contexts. The first is 'time point accuracy' that is measured at a specific future time point and is used to analyze the microscopic tendency. The second is 'over-all accuracy' that is measured as the average of the time point accuracies and is used to assess global performance.

The time point accuracy for all seven types of algorithms is shown in Fig. 4a; it is measured as the RMS of relative error, that is, the absolute error divided by the real value $| p_p - p_r | / p_r$, where p_p and p_r means predicted and ground truth (real) values, respectively, for three types of vegetables i. In Fig. 4b, the representative—the best three—results in the over-all accuracy described in Sect. 6.2.

There is a distinguished tendency of TATP of LSTM and GRU in the earlier weeks of the first third of a year before the 17th week, that is, they produce results with substantially high error rates at some time points, although they show high performance on average. In contrast, TATP of SRNN and all DFTSs perform mostly well on average. These results suggest that the gate mechanism in LSTM and GRU, that captures the direct relations between a specific time point in past to the current, requires sufficient duration and/or attentiveness in training.

Fig. 4 Time point accuracy of forecasted future time series

(a) Time point accuracy of all seven types of forecasting algorithms.
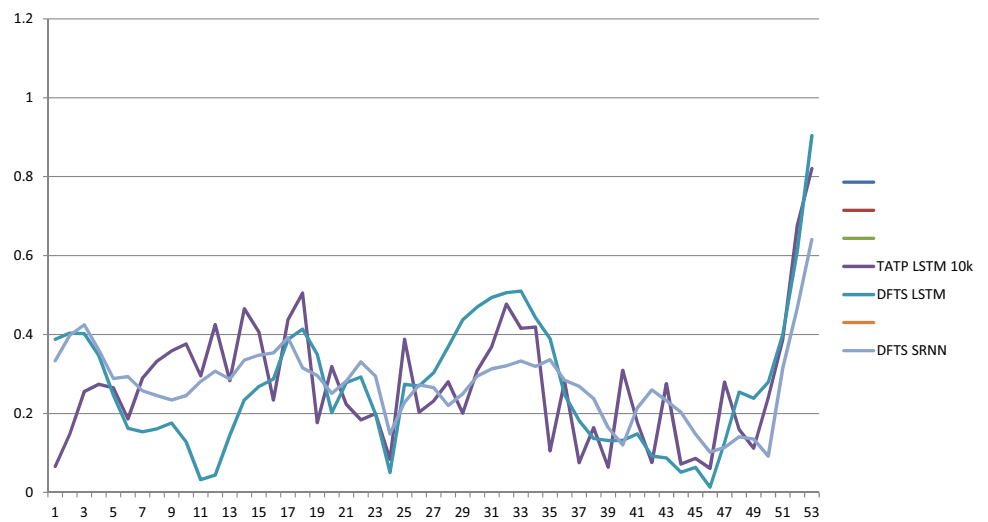
(b) Time point accuracy of TATP-LSTM-10k, DFTS-LSTM, and DFTS-SRNN.

Because DFTS forces the loss of all values to decrease equally, value changes in timeline are smoother than that of TATP.

6.2 Over-all accuracy

We used two kinds of error rates to evaluate the over-all accuracy (global performance) and to compare forecasted results with real-observed values. When calculating the over-all error by time point errors, RMS error is widely used. In addition, we use mean absolute (MA) error to estimate the results without over-estimation of RMS, i.e., amplifying the relatively bigger error by square calculation. As described in the previous section, relative error is used instead of absolute error. The definition of the error estimation is as follows:

root mean square percentage error (RMSPE);

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{f_i - y_i}{y_i}\right)^2}, \tag{3}$$

mean absolute percentage error (MAPE);

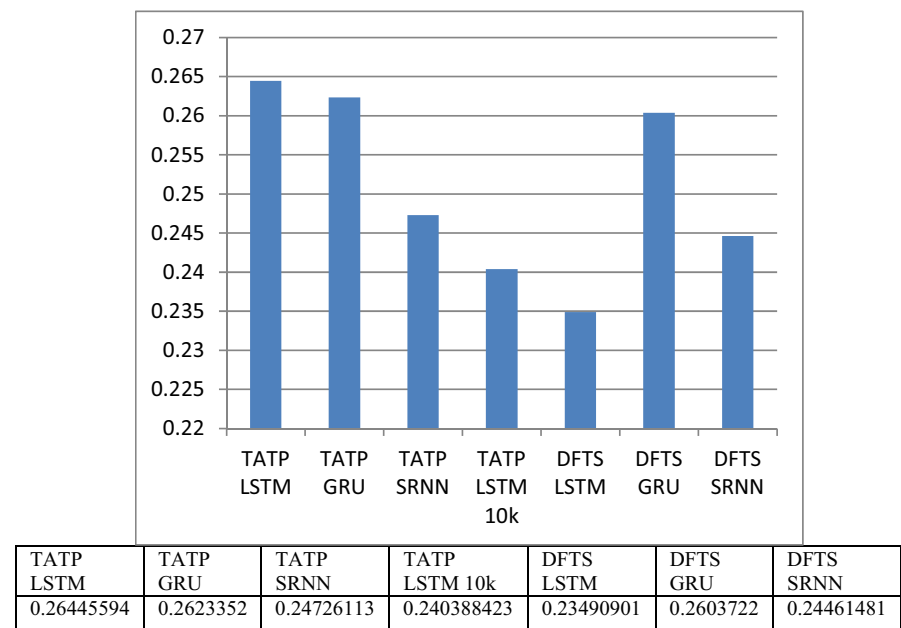$$\frac{1}{n}\sum_{i=1}^{n}\left|\frac{f_i - y_i}{y_i}\right|, \tag{4}$$

where f_i and y_i denote forecasted and real-observed values, respectively, and i denotes the number of time points of forecasted results. We omitted the multiplication by 100 that is used for expressing percentages. The error rates of the seven types of algorithms are shown in Fig. 5.

For TATP, LSTM-300 and GRU-200, which were not sufficiently trained, showed lower performance than the linear model (simple RNN) within 300 epochs. GRU did not show

Fig. 5 Over-all accuracy

TATP LSTM	TATP GRU	TATP SRNN	TATP LSTM 10k	DFTS LSTM	DFTS GRU	DFTS SRNN
0.34637597	0.36147548	0.31843328	0.313537357	0.31806301	0.36152939	0.29163083

(a) RMSPE of global performance.

TATP LSTM	TATP GRU	TATP SRNN	TATP LSTM 10k	DFTS LSTM	DFTS GRU	DFTS SRNN
0.26445594	0.2623352	0.24726113	0.240388423	0.23490901	0.2603722	0.24461481
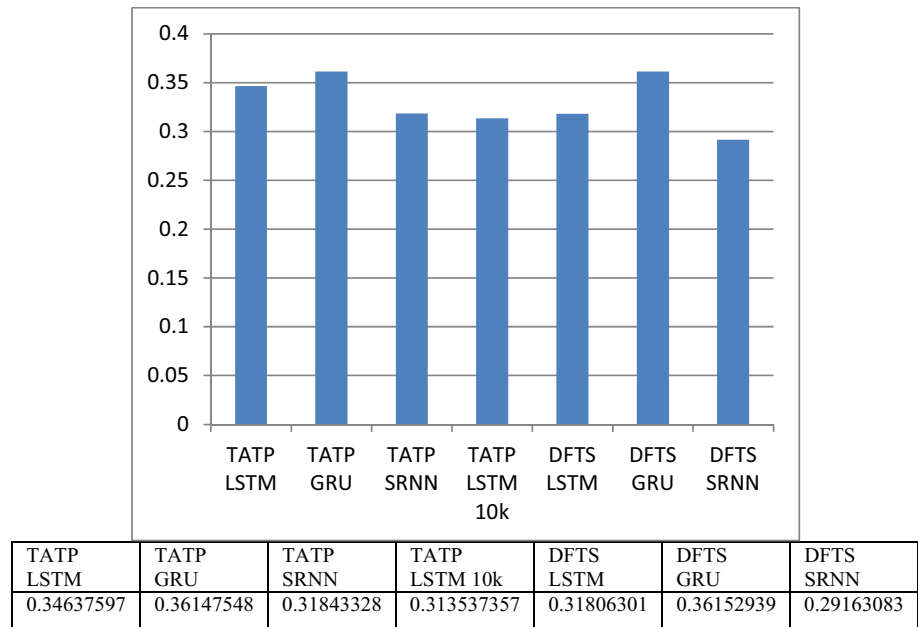
(b) MAPE of global performance.

improved performance even after deeper learning up to 10,000 epochs suggesting that it was over-fitted.

However, LSTM can be trained with higher number of epochs. TATP of LSTM within 10,000 epochs, a combination of the best time point forecaster selected among the 10,000 epochs, showed the highest global performance in the measurement of both RMSPE and MAPE.

For DFTS, when measured by RMSPE, SRNN showed a higher performance than that of both LSTM and GRU. The performance, however, differs significantly, when measured by MAPE. LSTM outperformed SRNN and GRU. This is because through RMSPE, a relatively higher error rate, which occasionally occurs, is amplified by the square calculation. In contrast, all the errors are equally averaged by MAPE. When LSTM is compared with SRNN, LSTM showed a higher performance on average than that of SRNN, but LSTM occasionally produced a high-error forecast than that of SRNN.

To conclude the evaluation based on accuracy, the global over-all accuracy was shown to be useful in evaluating the total performance of the algorithms, but we also need to establish the time point accuracy at the micro-level to investigate the details of the performance, especially when analyzing the microscopic trends.

We conclude the following regarding the evaluation on error rate (accuracy):

(1) LSTM, after enough training, shows a higher performance on average than that of SRNN both in the case of TATP and DFTS.
(2) LSTM seems to have a tendency of producing relatively larger error data than that of SRNN occasionally.
(3) DFTS shows a higher performance than that of TATP on average both in the case of LSTM and SRNN.
(4) In DFTS, LSTM can learn faster than TATP, that is, LSTM shows a high performance in smaller epochs of approximately 1000 epochs, which is the same duration needed for the training of SRNN and GRU.
(5) GRU shows a lower performance than that of LSTM and even SRNN in many cases.
(6) To evaluate the performance, MAPE has equal importance to that of RMSPE.
(7) The representative three algorithms (TATP LSTM-10k, DFTS LSTM, and DFTS SRNN) are the top three algorithms with the least error rates and they have a nearly equal ability of capturing global tendency, as shown in Fig. 4b.
(8) Comparing TATP and DFTS, DFTS always overcomes TATP for any basic forecaster when evaluated by MAPE. This fact matches the results reported in [23], i.e., multiple-output approaches (such as DFTS) are invariably better than single-output approaches (such as TATP). Our experiment suggests that their result could hold even for the neural networks with gate mechanism of LSTM and GRU.

7 Conservation of statistical characteristics

In this section, we describe another evaluation measure for forecasting algorithms in the context of the ability to conserve the statistical characteristics of the time series.

By conservation of statistical characteristics, we mean the ability of the time series forecasting algorithm to conserve the statistical characteristics of the original past time series in the forecasted future time series, or in other words, whether the same characteristics are observed in the generated future time series.

In social systems, including the market mechanisms, the time series observed in the systems does not always follow the Gaussian distribution. For instance, as shown in literature [27], many financial price movements in short-term trade do not follow the Gaussian distribution, but rather the Lévy stable distribution.

Compared with the Gaussian distribution, the Lévy stable distribution characterizes the nature of the target system more closely. For instance, market price changes slightly on usual days, but also occasionally changes dramatically. This tendency matches our feelings for the markets. In terms of mathematical features, (1) it has a sharp peak at the average value, and the probability is less than the Gaussian around the average value, and (2) it has a long or fat tail especially outside 3-sigma (three times the standard deviation) region, although the probability decreases rapidly outside 3-sigma in the Gaussian distribution.

Market price is the results of huge numbers of agent interactions in the society. The statistical characteristics of its PDF is expected to work as a kind of "finger print" of the target system, which we try to use as a way of evaluation.

First, we investigated the statistical characteristics of the original past time series that was used to train the forecasting algorithms. The data used was for a total of 302 weeks
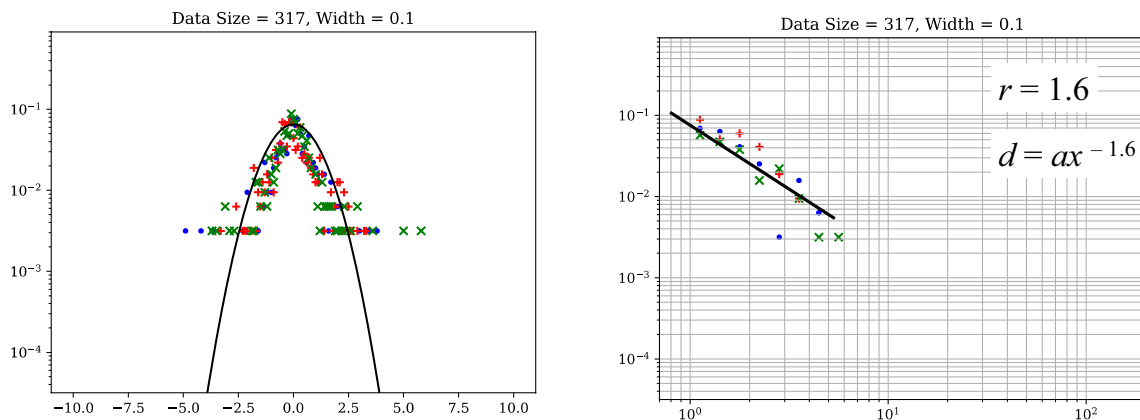

Fig. 6 PDF of real-observed past time series. **a** Left: linear-log plot. **b** Right: log–log plot

wherein the price data of all three kinds of products were simultaneously available.

Figure 6 shows the result of the analysis of the original past time series represented as a PDF. In Fig. 6a, the probability distribution is plotted in linear-log scale. Horizontal axis is the value change in the time series normalized by standard deviation, and the vertical axis is the existence probability of the value. A solid black line represents the Gaussian distribution for comparison.

From the findings, there are many data points outside the Gaussian curve especially outside 3-sigma region. In a Gaussian distribution, there is a very limited number of data points outside the 3-sigma region (≤ 0.27% of the data points). Additionally, the function seems to have a sharp peak at the average value, and less number of data exists around the average value. These findings show that the target time series follows the Lévy stable distribution rather than the Gaussian distribution.

To investigate further, we examined the log–log plot of the PDF (Fig. 6b). The Lévy stable distribution follows the power law that is represented by a straight line in a log–log plot outside approximately the 1-sigma region. Therefore, we can confirm the Lévy stable distribution by observing the log–log plot of the distribution. As shown in Fig. 6b, the data follow the power law with the coefficient 1.6. Although the linearity outside 3-sigma is broken, this might be caused by the limited number of samples. We conclude that the original past time series follows the Lévy stable distribution with a coefficient of 1.6.

Our objective is to assess whether the stylistic characteristics of the Lévy stable distribution was conserved by the transformation of the forecasting mechanism, that is, whether the generated future time series also followed the original Lévy stable distribution or not.

To examine the performance of conservation of statistical characteristics, we analyzed the PDF of the generated time series. For each forecasting algorithm, the best 10 forecasters, in order of accuracy, were selected. Each of the forecasters generated 53 weeks forecast results based on the ground truth (real) data observed for 54 weeks. Thus, a total of 28,620 data points of value differences were obtained. There is no problem with gathering the best 10
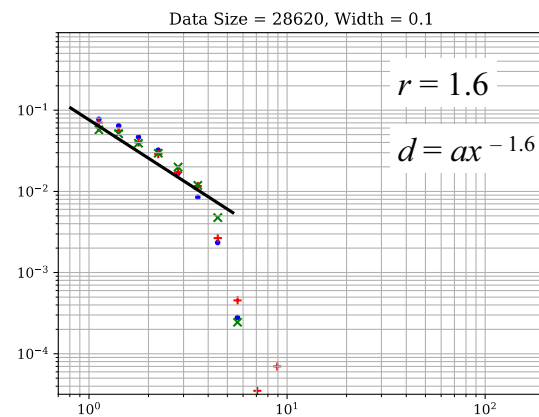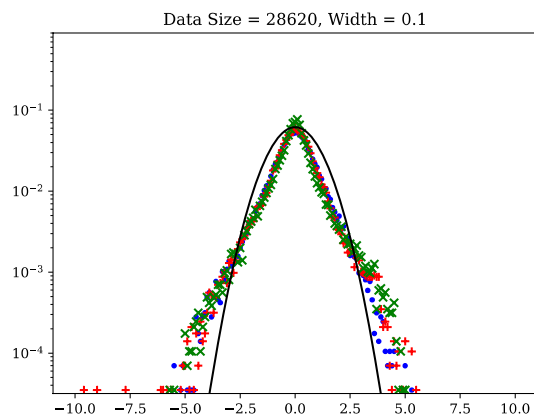

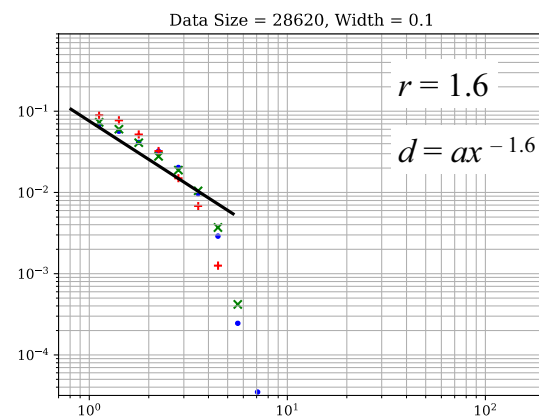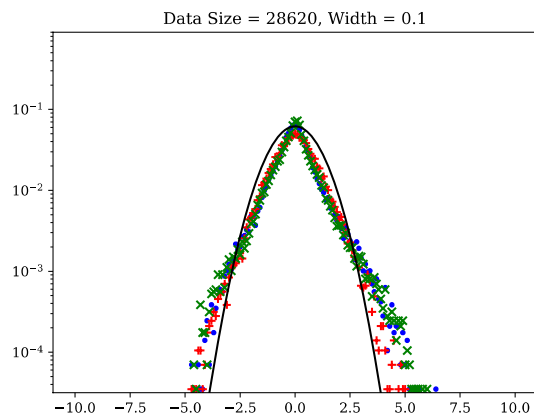**Fig. 7** PDF of time series by TATP of LSTM 300 epochs

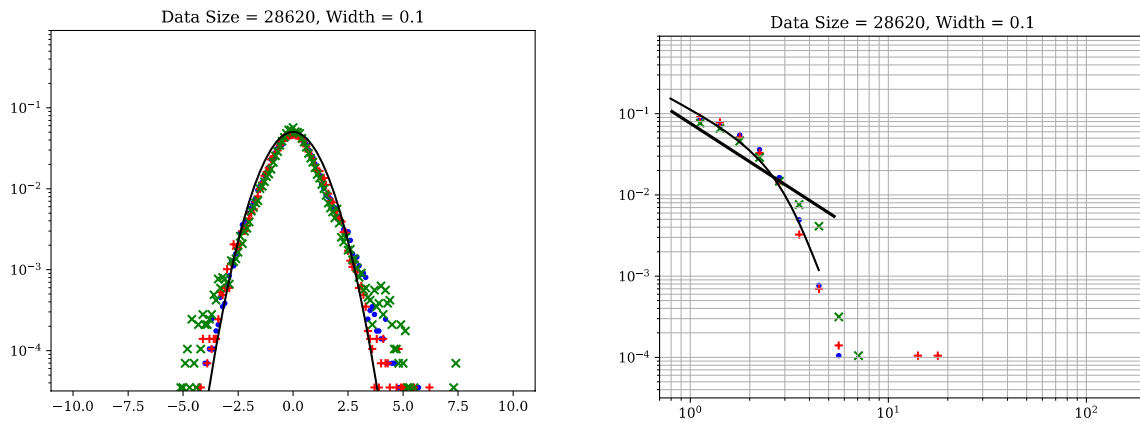**Fig. 8** PDF of time series by TATP of GRU 200 epochs

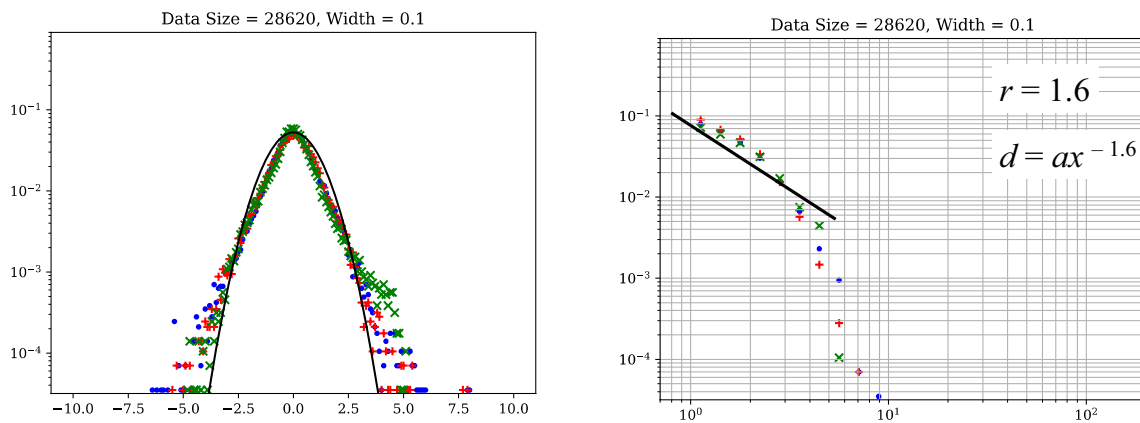**Fig. 9** PDF of time series by TATP of SRNN 300 epochs

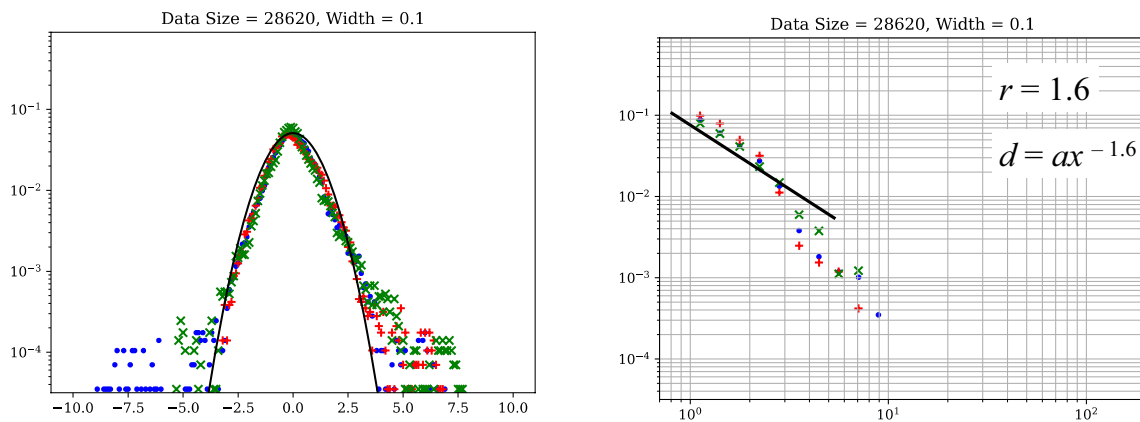**Fig. 10** PDF of time series by TATP of LSTM 10,000 epochs

Fig. 11 PDF of time series by DFTS of LSTM 1000 epochs

forecasters, because the obtained distribution automatically forms a stable distribution, when stable distributions (such as Lévy distribution) have been gathered and added [28].

The results for all seven algorithms are shown in Figs. 7, 8, 9, 10, 11, 12 and 13. Many observed future time series show non-Gaussian distribution characteristics, especially there are many observed data points outside the 3-sigma region.
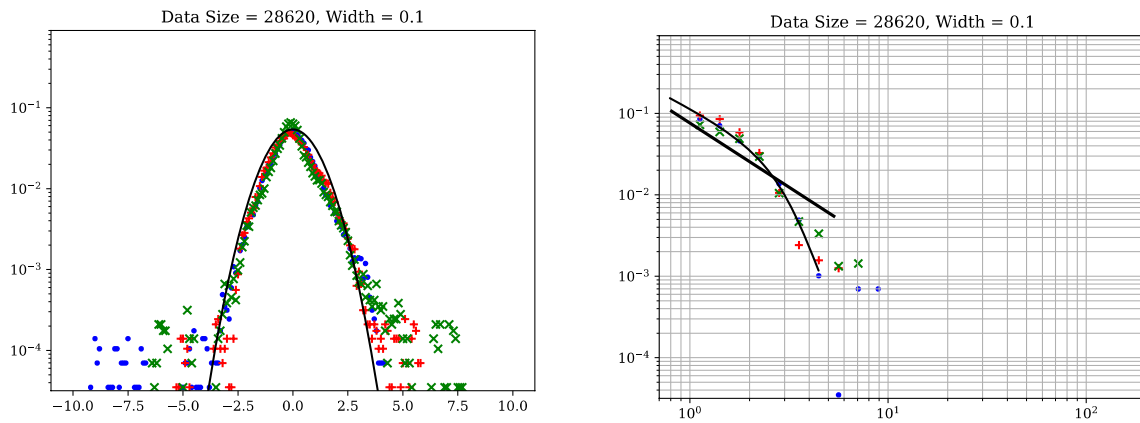
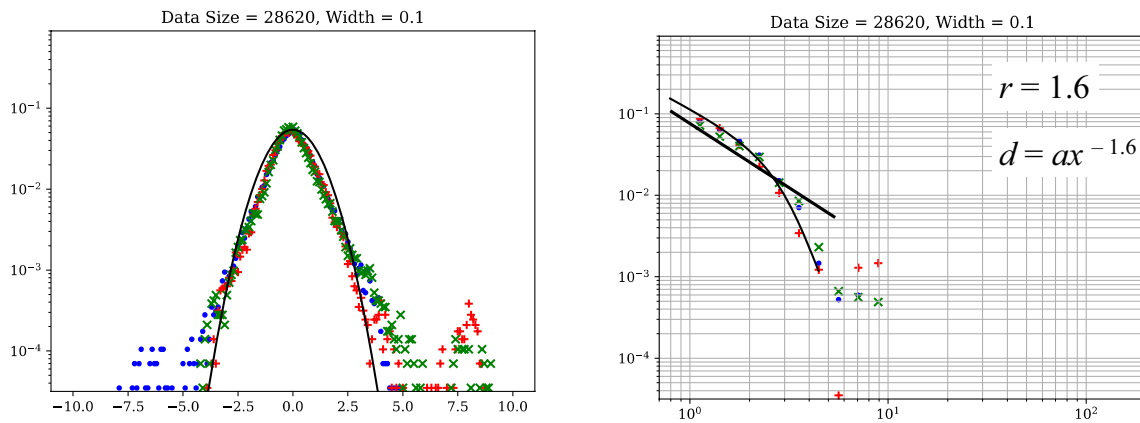**Fig. 12** PDF of time series by DFTS of GRU 1000 epochs

Fig. 13 PDF of time series by DFTS of SRNN 1000 epochs

$$r = 1.6$$

$$d = ax^{-1.6}$$

Fig. 14 Trends by seasonal decomposition when cycle is 53 weeks

In TATP of LSTM-300 and LSTM-10k (Figs. 7, 10), the distinct features of a Lévy stable distribution can be clearly observed in the linear-log plot, and the PDF can be represented as a straight line in the log–log plot with a coefficient 1.6 that is equal to the original past time series. TATP of GRU also conserves the statistical characteristics of the probability distribution (Fig. 8), but simple RNN seems not to have the ability to conserve the Lévy stable distribution feature as shown in Fig. 9. The generated sequence by simple RNN resembles a Gaussian distribution rather than a Lévy stable distribution.

For DFTS, all three types of algorithms, LSTM, GRU, and SRNN, show the characteristics of Lévy distribution, that is, a sharp peak at the average value, less probability density around the average value, and a long tail outside the 3-sigma region (Figs. 11a, 12a, 13a). The extent of the
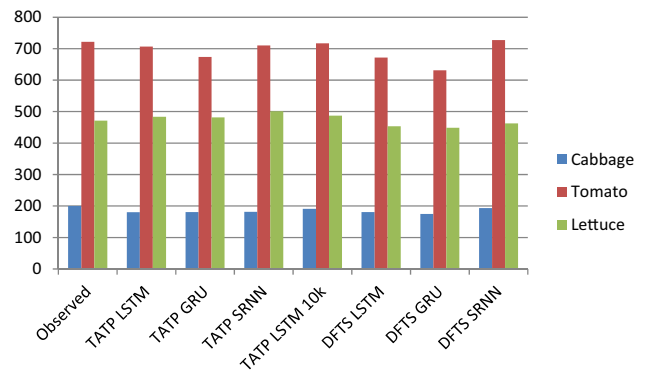
characteristics is, however, weaker in GRU and SRNN than in LSTM (Figs. 11b, 12b, 13b).

We have carried out another statistical analysis by seasonal decomposition. When taking 53 weeks (about 1-year) cycle for originally observed real values and forecasted values by our methods. The results are shown in Fig. 14. There

seem to be no significant differences between forecasting algorithms.

8 Comparison among algorithms

Seven kinds of forecasting algorithms show different behaviors in the three evaluation measures. The comparison of the behaviors of the forecasting algorithms can be summarized as follows.

(1) For TATP, the well-trained LSTM-10k shows superior performance in terms of accuracy, and it also seems to conserve the characteristics of the Lévy distribution better than SRNN-300 and GRU-200 forecasting algorithms.

(2) For DFTS, LSTM and SRNN show the highest performance in MA and RMS, respectively, in terms of accuracy. All three kinds of DFTS algorithms LSTM, GRU, and SRNN seem to conserve the statistical characteristics best in this order.

(3) To compare LSTM in TATP and DFTS, the accuracy of TATP is higher than that of DFTS when measured using RMS, and the accuracy of DFTS is higher than that of TATP when measured using MA. This implies that TATP tends to generate major errors occasionally. There is no significant difference in the conservation ability of statistical characteristics (Figs. 10b, 11b).

(4) When SRNN in TATP was compared with DFTS, DFTS shows higher performance than TATP both in the terms of accuracy (both in RMS and MA) and in the conservation ability of statistical characteristics. This conclusion is mathematically natural because SRNN with linear activation function is a polynomial function and DFTS puts an additional constraint among the variables, which helps to narrow the range of the variables.

To conclude the comparison study of algorithms, LSTM demonstrated a higher performance than that of GRU and SRNN based on the accuracy and the conservation ability of the statistical characteristics, but only when it is trained in the right direction with enough epochs, for the data set. The selection of TATP and DFTS differently influences the accuracy in the case of RMS and MA.

In contrast, SRNN behaves differently. When forecasted with a linear model of SRNN, it shows a higher performance both in accuracy and in statistical characteristic conservation when a constraint by DFTS on the values is added.

9 Discussion

Whether RNNs outperform conventional linear methods has been under discussion. In the previous studies, LSTM has been shown to outperform ARIMA for several kinds of financial indices [29]. In contrast, RNNs have also been indicated to show poorer performance than that of the conventional methods [30].

We believe that this difference is caused by the selection of evaluation measures and the wrong handling of the forecasting methods, especially the way of learning of RNNs. The evaluation measures need to distinguish which attribute of time series should be focused on, that is, time point accuracy, over-all accuracy, or statistical characteristics. In many previous reports, the measure of evaluation is fixed, and other attributes of the time series are ignored. We think this is one reason for the varying analytical results of RNNs and conventional methods for time series forecasting.

In addition, there is a possibility that RNNs were not sufficiently trained in the previous studies concluding that RNNs showed poorer performance than that of the conventional linear methods. As seen in Sect. 6, LSTM with less training epochs cannot show better performance than SRNN. It requires training with enough epochs in the right direction to become a high-performing forecaster both in TATP and DFTS.

Regarding the difference between TATP and DFTS, when applying SRNN, DFTS always performs better than TATP. This is because DFTS puts a constraint to the forecasted time series that forces the values of time series to change simultaneously. Since SRNN with linear activation is a polynomial function, this kind of constraint helps the forecasted time series to reduce errors.

In contrast, this type of constraint does not always work well for the non-linear methods such as LSTM and GRU. TATP shows a higher performance than that of DFTS in over-all accuracy when LSTM is applied. This is because in TATP, each forecaster for a specific point in future can concentrate on the specific time point, and it can pursue low forecast errors between generated and real observed values at the time point. This assumption seems to hold in this experiment.

To compare this research with other approaches, first, we provide a way to enable the selection of basic forecasting mechanism according to the specified criteria such as time-point accuracy, global accuracy, and statistical characteristics of generated future time series. Second, we clarified the effectiveness and limitation of controlling the influence using the gate mechanism. Gate mechanism seems to capture subtle relations embedded in time series, which possibly makes the accuracy and/

or statistical characteristics better or worse occasionally. Third, the algorithm fits the limitation of small numbers of training data with missing part of temporal duration and product. Actually, it is possible to forecast 1-year cycle (53 week) after just training by 20 weeks that is less than the half number of the whole cycle.

One shortcoming of our method is the extensibility to higher dimensional variables of target time series. In this paper, the target is 3-dimentional vector. When forecasting these three variables independently, the accuracy in RMSE sense decreases about 1 percent, which suggests that these three variables are almost uncorrelated. Exploiting correlations between variables in our framework is left for future work.

The proposed method can be applied to a field where the statistical aspects of predicted time series are important, for instance, to improve the simulation ability of economic systems [31, 32], transportation indirect control at the macro-level [33, 34], and remote-sensing ability under noisy environments in the internet of things (IoT) [35].

10 Conclusion

We have implemented forecasting tasks of commodity price based on RNNs. Two methods of generating future times series, TATP and DFTS, are proposed, each of which use a unique kind of RNN including LSTM, GRU, or simple RNN as the basic forecaster.

The results were examined based on the accuracy (error rate in RMS and MA) and the conservation performance of the statistical characteristics by conformity to the power law in the stable distribution such as Lévy distribution.

The comparison studies among the algorithms showed that LSTM shows the highest performance in terms of accuracy (low error rate) and the conservation performance of the statistical characteristic, only when it was well trained for enough epochs in the right direction. Simple RNN, as a generalized linear modeling method, shows better performance in error rate than GRU and LSTM with less numbers of learning epochs. LSTM after enough training, however, shows a better performance than the generalized linear methods of SRNN.

When comparing TATP with DFTS, DFTS introduces additional constraint that force forecasted future values to simultaneously approach the real-observed value. This constraint works well for linear methods (simple RNN), because a linear method is a polynomial of coefficients and past value variables, and the polynomial can be easily optimized by the additional constraint. In contrast, this constraint does not always work well when non-linearity exists in the system such as gate mechanism in LSTM and GRU. TATP of LSTM searches the optimized value in an independent manner among variables, and it shows better performance than DFTS based on the RMS error.

They were evaluated the forecasting algorithms based on the accuracy and the conservation ability of statistical characteristics. We found that LSTM, after sufficient training, shows a higher performance with regards to not only the error rate but also the conservation ability of a probability distribution compared to that of other RNNs, both in TATP and DFTS. These results suggest that the selection of the effective forecasting methods depends on the evaluation criteria and that the combinations of forecasting methods is useful depending on the intended application.

To conclude the comparison study, selecting the best algorithms depends on which aspects or attributes of the target systems we aim to focus on. The linear modeling methods, such as SRNN with linear activation function used in this study, can forecast the future sequence with fewer epochs meaning that low computational costs are needed. Non-linear methods such as LSTM require enough epochs of learning in the right direction implying that there are cases when LSTM fails to find good solutions even after long learning periods. However, once it is well trained, the non-linear methods (mainly demonstrating gate mechanism) show improved performance in accuracy.

The LSTM forecasting method also demonstrates high conservation performance of the statistical characteristics of stable distribution functions such as Lévy distribution.

Future work includes constructing forecasting mechanisms with rapid learning and preciseness by combining different types of basic forecasting methods. Second is the automatic selection of forecasting methods depending on the application need such as time-point accuracy, overall accuracy, and/or statistical characteristics of generated time series.

Compliance with ethical standards

Conflict of interest The author declares they have no conflict of interest.

References

1. Box G, Jenkins G (1976) Time series analysis: forecasting and control. Holden Day
2. Funahashi K, Nakamura Y (1993) Approximation of dynamical systems by continuous time recurrent neural networks. Neural Netw 6(6):801–806
3. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
4. Cho K, van Merrienboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using RNN encoder–decoder for statistical machine translation. Preprint arXiv:1406.1078
5. Faruk DO (2010) A hybrid neural network and ARIMA model for water quality time series prediction. Eng Appl Artif Intell 23(4):586–594
6. Kardakos EG, Alexiadis MC, Vagropoulos SI, Simoglou CK, Biskas PN, Bakirtzis AG (2013) Application of time series and artificial neural network models in short-term forecasting of PV power generation. In: Proceedings of 48th international universities' power engineering conference (UPEC), IEEE. https://doi.org/10.1109/upec.2013.6714975
7. Khashei M, Bijari M (2011) A novel hybridization of artificial neural networks and ARIMA models for time series forecasting. Appl Soft Comput 11(2):1491–2934
8. Lin CJ, Chen HE, Lee TS (2011) Forecasting tourism demand using time series, artificial neural networks and multivariate adaptive regression splines: evidence from Taiwan. Int J Bus Adm 2(2):14–24
9. Hajirahimi Z, Khashei M (2019) Weighted sequential hybrid approaches for time series forecasting. Phys A 531:121717
10. Binkowski M, Marti G, Donnat P (2018) Autoregressive convolutional neural networks for asynchronous time series. In: Proceedings of the 35th international conference on machine learning (ICML-18), pp 580–589
11. Neil D, Pfeiffer M, Liu S-C (2016) Phased LSTM: accelerating recurrent network training for long or event-based sequences. In: Proceedings of the 30th international conference on neural information processing systems (NIPS'16), pp 3889–3897
12. Hsu D (2017) Time series forecasting based on augmented long short-term memory. Preprint arXiv:1707.00666
13. Borovykh A, Bohte S, Oosterlee CW (2017) Conditional time series forecasting with convolutional neural networks. Preprint arXiv:1703.04691
14. Deng Y, Bao F, Kong Y, Ren Z, Dai Q (2017) Deep direct reinforcement learning for financial signal representation and trading. IEEE Trans Neural Netw Learn Syst 28(3):653–664
15. Salinas D, Flunkert V, Gasthaus J (2017) DeepAR: probabilistic forecasting with autoregressive recurrent networks. Preprint arXiv:1704.04110
16. Lai G, Chang W-C, Yang Y, Liu H (2018) Modeling long- and short-term temporal patterns with deep neural networks. In: Proceedings of the 41st international ACM SIGIR conference on research & development in information retrieval, ACM, pp 95–104
17. Yu H-F, Rao N, Dhillon IS (2016) Temporal regularized matrix factorization for high-dimensional time series prediction. In: Proceedings of 30th conference on neural information processing systems (NIPS)
18. Franceschi J-Y, Dieuleveut A, Martin J (2019) Unsupervised scalable representation learning for multivariate time series. Adv Neural Inf Process Syst 32:4650
19. Sen R, Yu H-F, Dhillon I (2019) Think globally, act locally: a deep neural network approach to high-dimensional time series forecasting. Adv Neural Inf Process Syst 32:4837–4846
20. Jha GK, Sinha K (2013) Agricultural price forecasting using neural network model: an innovative information delivery system. Agric Econ Res Rev 26(2):229–239
21. Zhang Y, Na S (2018) A novel agricultural commodity price forecasting model based on fuzzy information granulation and MEA-SVM model. Math Probl Eng 2018, 2540681
22. Varun R, Neema N, Sahana HP, Sathvik A, Mohammed M (2019) Agriculture commodity price forecasting using Ml techniques. Int J Innov Technol Explor Eng (IJITEE) 9(2S):729–732
23. Taieb SB, Bontempi G, Atiya A, Sorjamaa A (2011) A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition. Preprint arXiv:1108.3259
24. MAFF, Ministry of Agriculture, Forestry and Fisheries (2020) http://www.maff.go.jp/e/index.html http://www.maff.go.jp/j/zyukyu/anpo/kouri/index.html (**in Japanese**)
25. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. Preprint arXiv:1412.6980
26. Kurumatani K (2018) Time series prediction of agricultural products price based on time alignment of recurrent neural networks. In: Proceedings of 2018 17th IEEE international conference on machine learning and applications (ICMLA-18), IEEE, pp 81–88. https://doi.org/10.1109/icmla.2018.00020
27. Mantegna RN, Stanley HE (1995) Scaling behaviour in the dynamics of an economic index. Nature 376:46–49
28. Shintani M, Umeno K (2018) Super generalized central limit theorem: limit distributions for sums of non-identical random variables with power laws. J Phys Soc Jpn. https://doi.org/10.7566/jpsj.87.043003
29. Siami-Namini S, Tavakoli N, Namin AS (2018) A comparison of ARIMA and LSTM in forecasting time series. In: Proceedings of 2018 17th IEEE international conference on machine learning and applications (ICMLA-18), IEEE, pp 1394–1401. https://doi.org/10.1109/icmla.2018.00227
30. Krauss C, Do XA, Huck N (2016) Deep neural networks, gradient-boosted trees, random forests: statistical arbitrage on the S&P 500, FAU Discussion Papers in Economics 03/2016, Friedrich-Alexander, University Erlangen-Nuremberg, Institute for Economics
31. Namatame A, Terano T, Kurumatani K (2002) Agent-based approaches in economic and social complex systems. IOS Press, Amsterdam
32. Kurumatani K, Kawamura H, Ohuchi A (2005) Market microstructure analysis by multiagent simulation in X-economy: comparison among technical indices. Inf Sci 170(1):65–74
33. Yamashita T, Izumi K, Kurumatani K (2004) Effect of using route information sharing to reduce traffic congestion. In: Kurumatani K, Chen S-H, Ohuchi A (eds) Multi-agent for mass user support, lecture notes in artificial intelligence (LNAI) 3012, Springer, Berlin, pp 86–104, ISBN 3-540-21940-4
34. Yamashita T, Izumi K, Kurumatani K, Nakashima H (2005) Smooth traffic flow with a cooperative car navigation system. In: Proceedings of forth international joint conference on autonomous agents and multi agent systems (AAMAS-05, Utrecht, Netherlands), pp 478–485

35. Sashima A, Kawamoto M, Kurumatani K (2018) A peer-to-peer telecare system using smart watches and wireless biosensors. Health Technol 8(5):317–328. https://doi.org/10.1007/s12553-018-0240-8