A Novel Hybrid STL-Based Model for Egg Price Forecasting

Liyun Mo[1], Minlan Jiang[1,2(✉)], Xiaosheng Fang[1], and Xiaowei Shi[3]

[1] College of Physics and Electronic Information Engineering, Zhejiang Normal University,
Jinhua, China
{xx99,webserver}@zjnu.cn

[2] Zhejiang Key Laboratory of Optical Information Detection and Display Technology Research,
Jinhua, China

[3] Hangzhou Hikvision Digital Technology Co., Hangzhou, China
shixiaowei@hikvision.com

Abstract. The egg industry is a significant contributor to the economy and requires a stable egg price for its sustainable growth. Accurate egg price prediction is crucial to monitor the market, provide reference for decision-making, and achieve early warning. This study presents a novel egg price forecasting model that combines the seasonal-trend decomposition based on loess (STL), temporal convolutional network (TCN), gated recurrent unit (GRU), and random forest (RF) methods to capture the nonlinear, seasonal, and cyclical characteristics of egg price series. The egg price series is decomposed into trend, seasonal, and residual components using the STL method. Select the model with the best prediction results for the single decomposition component. So these components are then predicted using the TCN, GRU, and RF models, respectively. The predicted values are then aggregated to form the final forecast. The empirical results demonstrate that the proposed hybrid model achieves the best performance, and compared to the best predicted single model, MSE, RMSE, MAE and MAPE were reduced by 66.35%, 41.94%, 30.08% and 29.81% respectively, and $\boldsymbol{R}^{\mathbf{2}}$ was improved by 3.48%. This study provides a promising alternative approach for egg price forecasting, and the results have implications for the development of price forecasting techniques for other agricultural products.

Keywords: egg price forecasting · seasonal-trend decomposition procedure based on loess · temporal convolutional network · gated recurrent unit insert · random forest

1 Introduction

The consumption of eggs constitutes a vital component in the daily protein intake of individuals, thereby contributing to the enhancement of the dietary patterns of the general populace. The availability of eggs at reasonable prices is crucial for safe-guarding the economic interests of both egg producers and operators, as well as satisfying the dietary

D. Kumar et al. (Eds.): IEIT 2023, AHSSEH 10, pp. 365–382, 2023.
https://doi.org/10.2991/978-94-6463-230-9_44

requirements of the public. Hence, the capability to forecast changes in egg prices accurately and in a timely manner is imperative for the sustained and robust growth of the egg market.

The fluctuation of egg prices has been well documented and exhibits cyclical, seasonal, and irregular volatility [1]. The underlying factors that drive these fluctuations include the growth in consumer demand, the increase in production costs, and the overall rise in price levels. Seasonal fluctuations, on the other hand, stem from the dynamic interplay between supply and demand, while irregular fluctuations result from the convergent effects of external factors such as pandemics, political actions, trade dynamics, and natural disasters [2].

The existing forecasting models can be classified into three categories: statistical models, machine learning (ML) models, and hybrid models. Statistical models were early adopters of predictive models. Such as autoregressive comprehensive moving average (ARIMA), but they are limited in their ability to forecast price data with complex characteristics. Of these categories, ML techniques have been reported to demonstrate successful outcomes, particularly in areas such as derivatives pricing, risk management, and market prediction [3]. In recent years, and as an important branch of machine learning, the deep learning (DL) has emerged as a promising tool for predictions, owing to its efficacy in various fields. When compared to traditional methods, DL is capable of capturing the nonlinear relationships present in price data with greater accuracy. A single model is often appropriate for data with a single characteristic, such as data series with typical linear, exponential, or periodic pat-terns. However, it may struggle to handle multiple characteristics contained in price series. Studies have shown that hybrid prediction approaches can improve fore-casting performance by addressing the limitations of single models, reducing un-certainty, and enhancing generalization capabilities simultaneously. In general, machine learning models generally predict better than statistical models, and hybrid models predict better than single machine learning models.

Various regression methods are used as traditional statistical methods. Models such as ARIMA. Yifan Zhang and Meihua Fan [4] forecast the egg prices in 500 County Markets in China using ARIMA. The predicted result is compared to the true value with a MAPE of 4.38%. Using time series analysis, Du Xiya [5] developed a time series model based on egg prices and predicted their egg prices for the next four months, and the results showed that the better-fitting ARIMA (2, 2, 2) model was able to predict the monthly average price trend of eggs better. Studies on agricultural price forecasting using statistical methods can handle general linear problems but have the disadvantage that the performance is not stable for non-linear price series.

Machine learning and deep learning based algorithms are new approaches to solving time series forecasting problems. These methods have been found to pro-duce more accurate results than traditional regression models. In recent years, learning models have been increasingly used to predict prices as price volatility in agricultural commodities has increased. Gao Yang and Ansebo [6] introduced a back-propagation neural network model into their study of egg price forecasting, which proved to have a lower forecast error than futures prices, with a minimum mean absolute percentage error value of 7.90%. The general regression neural network [7] and grey models [8] have also been applied to egg price forecasting, but a survey of the literature shows that grey models are suitable for exponential series and less suitable for egg price forecasting. The long short-term memory (LSTM) model is the most used model in the field of price prediction. For example, Liu Xue [9] and Wisdom Yan [10], applied the LSTM to egg price prediction, and the results showed that the LSTM prediction effect is better than ARIMA and SVR.

While a single model can predict prices, much of the literature shows that hybrid models are more advantageous in their predictions [11]. Many hybrid models first decompose the original sequence, which encompasses multivariate features, into a few distinct components that are more regular and easier to predict [12]. These components are then subject to time series analysis techniques or machine learning methods for forecasting separately. The resulting forecasts are then aggregated to achieve a more comprehensive forecast. The STL-based hybrid models have demonstrated promising potential as an alternative. The STL decomposition method breaks down the original price series into its trend, seasonal, and residual terms, which represent various typical characteristics that are more straightforward to predict. The trend term encompasses cyclical fluctuations and the potential growth rate, while the residual term encompasses irregular fluctuations.

Given these characteristics, egg price series present as typical nonlinear and nonstationary time series that pose a challenge for prediction. Given that the primary drivers of egg prices, including consumer behavior, supply and demand, disease, and policy, are difficult to quantify and predict, this study adopts the use of historical egg prices as the sole basis for prediction. This approach is premised on the under-standing that all of these factors are reflected in past egg prices and can be used to inform predictions.

The above survey shows that, in general, hybrid models predict best, then single machine learning models, and finally statistical models. So this study presents a novel STL-TCN-GRU-RF hybrid model for forecasting egg prices. The following are the main contributions of this research: 1) A new method for estimating short-term egg prices is developed. The training series is decomposed using the STL into its cyclical-trend, seasonal, and residual components, which are then used for forecasting. 2) In this paper, we select the model with the best prediction for each component separately by comparing the experiments to achieve the best prediction results. Thus, the cyclical-trend component, residual component, and seasonal component are predicted using the TCN, GRU and RF models, respectively. The resulting forecasts are then aggregated to produce the final egg price forecasts.

2 Materials and Methods

2.1 Seasonal-Trend Decomposition Procedure Based on Loess

The seasonal-Trend decomposition procedure based on Loess was introduced by Cleveland R as a time series filtering method [13]. This method aims to decompose a time series into three additive components: the trend, seasonal, and residual terms. The STL is considered to be more robust compared to traditional methods, such as the Hodrick-Prescott filter and X-12-ARIMA, in handling outliers in processed time series. The STL decomposes a given egg price series, represented by Y_t, where t= 1, 2,…,n denotes time, into trend T_t, seasonal S_t, and residual R_t components: $Y_t = S_t + T_t + R_t$. The residual component is commonly used to represent irregular fluctuations. The STL methodology comprises of two iterative procedures, referred to as the inner loop and the outer loop.

The inner loop performs the seasonal smoothing and trend smoothing, which result in the updating of the seasonal component and the trend component respectively, in a single iteration. The inner loop consists of six distinct steps as follows:

Step 1: Detrending. In the $(k + 1)^{th}$ iteration of the inner loop, the original series Y_t is detrended using the estimated trend component $T_t^{(k)}$ from the k^{th} pass, resulting in the detrended series $Y_t^{detrend} = Y_t - Y_t^{(k)}$;

Step 2: Seasonal Smoothing. The detrended series $Y_t^{detrend}$ is smoothed using a Loess smoother to obtain a preliminary seasonal component $\tilde{S}_t^{(k+1)}$;

Step 3: Low-Pass Filtering of Smoothed Seasonality. The preliminary seasonal component $\tilde{S}_t^{(k+1)}$ is further processed with a low-pass filter and a subsequent Loess smoother, to obtain a residual trend component $\tilde{T}_t^{(k+1)}$;

Step 4: Detrending of Smoothed Seasonality. The seasonal component $S_t^{(k+1)}$ is obtained as the difference between the low-pass filtered values and the preliminary seasonal component, that is $S_t^{(k+1)} = \tilde{S}_t^{(k+1)} - \tilde{T}_t^{(k+1)}$;

Step 5: Deseasonalizing. The original series Y_t is reduced by the seasonal component $S_t^{(k+1)}$ to obtain the deseasonalized series $Y_t^{deseason} = Y_t - S_t^{(k+1)}$;

Step 6: Trend Smoothing. The deseasonalized series $Y_t^{deseason}$ is smoothed using a Loess smoother to obtain the trend component $T_t^{(k+1)}$.

Upon the completion of the inner loop, the decomposition of the egg price time series into seasonal component S_t, trend component T_t, and residual component R_t via (1) in the outer loop.

$$R_t^{(k+1)} = Y_t - S_t^{(k+1)} - T_t^{(k+1)} \tag{1}$$

2.2 Temporal Convolutional Network

The trend component of the egg price series represents long-term cyclical variation in a non-linear and non-smooth time series. Temporal convolutional networks [14] are a modified convolutional structure that can adapt to the temporal nature of temporal data, and can provide a field of view for temporal modelling, consisting of causally dilated convolutions and residual blocks that can change the perceptual field by changing hyperparameters. With the advantages of a simple structure and support for parallel computing, it has more applications in load and wind prediction and anomaly detection, and has been shown to outperform LSTM in a variety of applications [15].

The causal convolution operates by predicting the value at time step $t + 1$ based on historical data up until time step t [16]. For instance, in the scenario where an input sequence $X = \{x_0, x_1, \ldots, x_T\}$ is to be forecasted, resulting in an output sequence $Y = \{y_0, y_1, \ldots, y_T\}$, the causal convolution is represented by the following:

$$y_t = \sum_{i=1}^{k} (w_i \cdot x_{t-k+i}) + b \tag{2}$$

where $w = w_1, w_2, \ldots, w_k$ is the convolution kernel, b is the bias term. To prevent the effects of gradient explosion and overfitting, TCN uses a dilation convolution approach.

Dilated convolution utilizes interval sampling at the input, as depicted in Fig. 1(a), enabling the size of the effective window to grow exponentially with the increasing number of layers. As a result, the size of the receptive field, represented by $\mathrm{RF_L}$, can be expressed as follows:

$$\mathrm{RF_L} = 1 + (\mathrm{k} - 1) \cdot \sum\nolimits_{\mathrm{i}=0}^{\mathrm{L}-1} \mathrm{d_i} \tag{3}$$

where k is the filter size, L is the layer of the convolution. The dilation factors of the $\mathrm{i^{th}}$ layer convolution, denoted as $\mathrm{d_i}$. The dilated convolution operation, denoted as F, on element *s* of a sequence is formally defined as follows:

$$\mathrm{F(s)} = (\mathrm{X} *_{\mathrm{d}} \mathrm{f})(\mathrm{s}) = \sum\nolimits_{\mathrm{i}=0}^{\mathrm{k}-1} \mathrm{f(i)} \cdot \mathrm{X_{s-d\cdot i}} \tag{4}$$

The input time series is represented by the vector X. The convolution operation is denoted by the symbol *. The convolution kernel is represented by the function f: 0, 1,…,k − 1. The term s - d · i accounts for the temporal direction of the past. Figure 1(a) shows the structure of the causal dilated convolution with the dilation factor d = 1, 2, 4 and the convolution kernel size k = 3.

The utilization of a residual connection addresses the issue of gradient instability [17]. The residual block in TCN consists of two modules, each comprising of causal dilated convolution, weight normalization, an activation function, and dropout. The overall structure of this configuration is presented in Fig. 1(b). Furthermore, an auxiliary 1 × 1 convolution is employed to ensure that elements of equal shape are summed in order to resolve the problem of differences in dimensionality between the original input and the output of the convolution layer. The output of the convolution layer is then summed with the original input, which is subsequently transformed by the activation function G before being outputted. This computation process is depicted in (5):

$$\mathrm{y} = \mathrm{G}(\mathrm{x} + \mathrm{f(x)}) \tag{5}$$

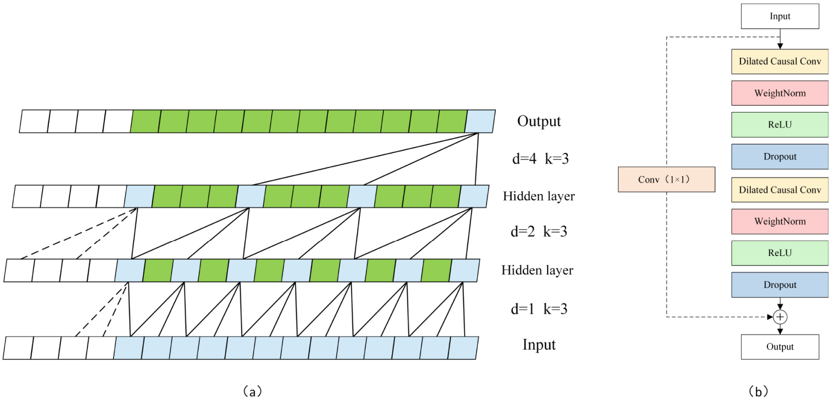


Fig. 1. The structural elements in the TCN: (a) are dilated causal convolutions, and (b) are residual blocks.

2.3 Gated Recurrent Unit

GRU is a variant of LSTM with a distinct structure for its hidden layer cell [18]. As discussed in the literature, the hidden layer cell of GRU consists of two gates, the reset gate r and the update gate z. Unlike LSTM, GRU eliminates the cell state and output gates and instead merges the input and forget gates of LSTM into a single update gate. This results in a simplification of the LSTM model structure and a reduction in network complexity as the information is directly transmitted through the hidden layer state h_t. The structure of the GRU hidden layer cell is illustrated in Fig. 2.

The calculation of the moment t for the forward computation in GRU is defined as follows:

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \tag{6}$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \tag{7}$$

$$\hat{h}_t = \sigma(W_h \cdot [r_t \cdot h_{t-1}, x_t] + b_h) \tag{8}$$

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \hat{h}_t \tag{9}$$

$$y_t = \sigma(W_o \cdot h_t + b_o) \tag{10}$$

where z_t represents the state of the update gate at time t, r_t represents the state of the reset gate at time t, $\hat{h}_t$ represents the candidate hidden layer state at time t, h_t represents the hidden layer state at time t, x_t represents the input vector for the GRU unit at time t, y_t represents the output of the GRU unit at time t, and W_z, b_z, W_r, b_r, W_h, b_h, W_o, b_o correspond to the weight matrices and coefficient vectors for the update gate, reset gate, candidate hidden layer, and output layer respectively.

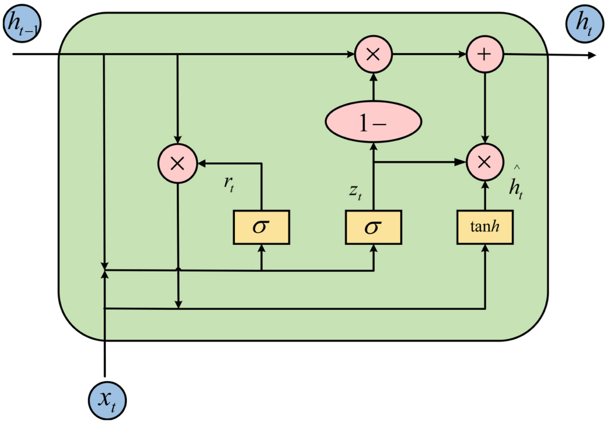


Fig. 2. Gated recurrent unit cell.

In (7), the value of r_t is determined by the sigmoidal activation function, producing a range of values between 0 and 1. The reset gate r_t plays a crucial role in the calculation of the candidate hidden layer state $\hat{h}_t$ in (8). When r_t approaches 1, more weight is given to retaining information from the previous hidden layer h_{t-1} in $\hat{h}_t$, and conversely, as r_t approaches 0, $\hat{h}_t$ is more likely to ignore h_{t-1}. This reset mechanism allows GRU to discard irrelevant information from previous moments and produce a more appropriate candidate hidden layer state. The update gate z_t in (9) controls the balance between retaining information from h_{t-1} and incorporating information from $\hat{h}_t$ in the output h_t. Finally, the output of the GRU unit y_t is calculated through (10).

2.4 Random Forest

The residual from STL decomposition is the residual left after removing trend and seasonality components, which captures the irregular fluctuations in egg prices resulting from external factors such as epidemics, policies, and disasters. As a supervised ensemble learning algorithm, RF is used to tackle high-dimensional classification and regression problems in variables [19]. RF operates by randomly selecting attributes to build multiple decision trees. Each tree is trained through bootstrapped sampling, meaning that a subset of the training data is randomly sampled with replacement to construct each tree. By utilizing a random subset of the attributes at each node in the decision tree, RF can avoid overfitting and improve generalization performance. In addition, the prediction results of all trees are aggregated by averaging to produce the final prediction of the RF model. Figure 3 presents the schematic illustration of RF.

During the training phase of the random forest algorithm, bootstrapping is utilized to create multiple sub-training datasets from the residual component. These sub-training datasets are then employed to train multiple decision trees in succession. In the prediction phase, the individual predictions generated by the decision trees are consolidated by taking the average, yielding the final prediction of the residual component.

2.5 Overall Process of the Proposed Hybrid Model

Based on the identified nonlinear, seasonal and cyclical characteristics of egg prices, this study proposes a hybrid model, which combines the STL, TCN, GRU and RF methods. The proposed model starts by decomposing the egg price series into three components, namely the trend component, seasonal component, and residual component, using the STL model. Subsequently, the TCN model is employed to forecast the trend component, the GRU model is used to forecast the seasonal component, and the RF method is used to forecast the residual component. The final egg price prediction is obtained by aggregating the predictions made for each of the three components. The overall process of the proposed model is depicted in Fig. 4.

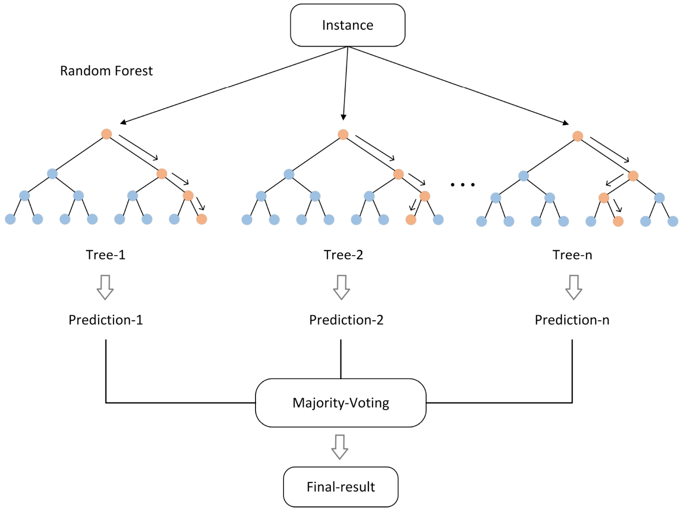


Fig. 3. Structure of RF.

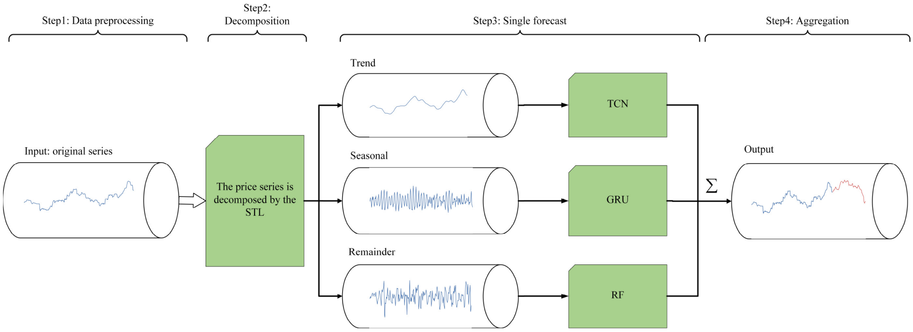


Fig. 4. Structure of the proposed hybrid model.

3 Experimental Results and Comparative Analysis

3.1 Data Source and Preprocessing

The experimental data for the study were obtained from daily average egg prices in China as recorded by Big Data for Agriculture and Rural Areas [20]. The data collected covered the time period from 1st January to 31st December 2022. Missing values in the data were estimated using the ARIMA model, yielding a total of 365 data points. The relevant statistical indices of the dataset are presented in Table 1 and are expressed in units of RMB/500g. As evidenced by Table 1, the egg price data exhibit high frequency fluctuations, with prices ranging from a high of 12.22 RMB/kg to a low of 7.46 RMB/kg, a fluctuation of 64%. This highlights the importance of accurate forecasting of egg prices in this context.

Table 1. Description of dataset

Data	Sample size	Maximum	Minimum	Mean	Standard deviation
Egg price	365	6.11	3.73	4.895	0.611

The generation of data samples was carried out using the sliding window technique. The samples were then split into two groups with 70% selected for training and 30% for testing through a series of experiments.

To enhance the convergence efficiency of the network, the sample data underwent Min-Max normalization, which transforms the data to the range [0,1] through the application of the following formula:

$$\mathrm{x}' = \frac{\mathrm{x} - \mathrm{x}_{\min}}{\mathrm{x}_{\max} - \mathrm{x}_{\min}} \tag{11}$$

where: x represents the original sample value, $\mathrm{x}_{\min}$ represents the minimum value of the sample, and $\mathrm{x}_{\max}$ represents the maximum value of the sample. The normalised value is represented by $\mathrm{x}^{'}$. To derive the actual predicted value and facilitate comparison with the actual set, the normalised output, $\mathrm{x}^{'}$, can be transformed back to x using the following formula:

$$\mathrm{x} = \mathrm{x}'(\mathrm{x}_{\max} - \mathrm{x}_{\min}) + \mathrm{x}_{\min} \tag{12}$$

3.2 Parameter Setting

All experiments were conducted using Python 3.8. The STL method was implemented through the STL functions of the statsmodels library. The selected competitive models such as SVR and RF were developed using the scikit-learn library. The TCN, GRU etc. were implemented on the TensorFlow backend using the Keras package.

The final parameter settings for the STL-TCN-GRU-RF model in this study are presented in Table 2. The default parameters were used for the STL method, with the exception of the period. The other parameters of the GRU are the same as those of the TCN.

3.3 Performance Criteria

In order to assess the forecasting efficacy of the STL-TCN-GRU-RF model, a suite of performance evaluation metrics was chosen. These metrics included mean square error (MSE), root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and coefficient of determination (R^2). A lower value for MSE, RMSE, MAE, and MAPE indicate a higher prediction accuracy, while a closer value of R^2 to 1 signifies a better model fit. The formulas for these metrics are as follows:

$$\mathrm{MSE} = \frac{1}{\mathrm{n}}\sum\nolimits_{\mathrm{i}=1}^{\mathrm{n}} (\hat{y}_i - y_i)^2 \tag{13}$$

Table 2. Parameter setting

Model	Parameter	Value
STL	Period	7
TCN	Window size	12
	Kernel size	2
	Filter numbers	12
	Dilation factor	[1,2,4,8]
	Optimizer	Adam
	Learning rate	0.001
	Loss	Mae
	Metrics	Accuracy
	Batch size	4
	Epochs	100
GRU	Layers	2
	Units	64
RF	N_estimators	100
	Max_depth	10
	Min_samples_split	2
	Min_samples_leaf	2

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{n}} \sum\nolimits_{\mathrm{i}=1}^{\mathrm{n}} (\hat{y}_i - y_i)^2} \tag{14}$$

$$\mathrm{MAE} = \frac{1}{\mathrm{n}} \sum\nolimits_{\mathrm{i}=1}^{\mathrm{n}} \left|\hat{y}_i - y_i\right| \tag{15}$$

$$\mathrm{MAPE} = \frac{1}{\mathrm{n}} \sum\nolimits_{1}^{\mathrm{n}} \left|\frac{\hat{y}_i - y_i}{y_i}\right| \times 100\% \tag{16}$$

$$\mathrm{R}^2 = 1 - \frac{\frac{1}{\mathrm{n}} \sum_{\mathrm{i}=1}^{\mathrm{n}} (\hat{y}_{\mathrm{i}} - y_{\mathrm{i}})^2}{\frac{1}{\mathrm{n}} \sum_{\mathrm{i}=1}^{\mathrm{n}} (\overline{y} - y_{\mathrm{i}})^2} \tag{17}$$

where n represents the total number of samples, $\hat{\mathrm{y}}_\mathrm{i}$ is denoted as the predicted value of the i^{th} sample, y_i is represented as the actual value of the i^{th} sample, and $\overline{\mathrm{y}}$ stands for the mean of the n samples.

3.4 Predictions of Egg Price Series

The proposed hybrid model was compared with several commonly utilized models for time series prediction, including single and combined models. Additionally, the prediction performance of the trend, season, and residual components were also compared with their respective counterparts.

1) **Predictions of Trend Components**.

The egg price time series is decomposed using the seasonal-Trend decomposition of time series method, and the trend component constitutes a significant portion of the egg price series, while the seasonal component exhibits periodic behavior. Conversely, the residual component does not present any discernible pattern.

The trend component prediction was evaluated by comparing the TCN with several commonly used single models. The statistical results of the trend series predictions are presented in Table 3.

From the Table 3, TCN emerges as the most effective model, followed by BP and SVR. A comparison between TCN and the second-best model reveals a significant improvement, with an average reduction of 48.58% in MSE, 28.1% in RMSE, 25.51% in MAE, and 25.34% in MAPE. The coefficients of determination for TCN exceed 0.99, indicating a high level of accuracy in its predictions and a strong fit between the predicted trend values and the true values. As a result, TCN is selected as the model for predicting trend components.

2) **Predictions of Seasonal Components.**

The prediction results for the seasonal component of egg prices are presented in Table 4. It should be noted that the MAPE metric is not included in this table as the seasonal component tends to fluctuate around 0, making this metric inapplicable.

As evident from Table 4, GRU emerged as the most effective model in predicting the seasonal components, followed by BP and SVR. In comparison to the second-best model, GRU demonstrated a reduction in average MSE, RMSE, and MAE by 43.59%, 24.72%, and 12% respectively. The coefficients of determination for GRU exceeded

Table 3. Statistical results of the predictions for the trend component

Model	**MSE**	**RMSE**	**MAE**	**MAPE**	R^2	**Rank**
TCN	0.0013	0.0358	0.0304	0.5625	0.991	1
BP	0.0025	0.0497	0.0409	0.7534	0.9825	2
SVR	0.0035	0.0588	0.0478	0.8829	0.9756	3
GRU	0.0035	0.0595	0.0503	0.8944	0.9751	4
RF	0.0157	0.1253	0.0842	1.5095	0.8894	5
Adaboost	0.0179	0.1339	0.0935	1.6773	0.8736	6
LightGBM	0.0218	0.1475	0.1086	1.9497	0.8467	7
GBDT	0.0233	0.1527	0.105	1.8828	0.8355	8
XGBoost	0.0286	0.1691	0.1166	2.0956	0.7985	9
RNN	0.0361	0.1901	0.177	3.2344	0.7452	10
Knn	0.0641	0.2533	0.2133	3.9351	0.5478	11
LSTM	0.0831	0.2882	0.2479	4.575	0.4143	12

Table 4. Statistical results of the predictions for the seasonal component

Model	MSE	RMSE	MAE	R^2	Rank
GRU	0.0002	0.0149	0.0113	0.9133	1
BP	0.0004	0.0198	0.0128	0.847	2
SVR	0.0004	0.0204	0.0135	0.8369	3
LSTM	0.0005	0.0226	0.0138	0.7993	4
RNN	0.0006	0.0236	0.0147	0.7819	5
TCN	0.0007	0.0257	0.0161	0.7407	6
LightGBM	0.0008	0.0287	0.0184	0.6778	7
Adaboost	0.0008	0.0288	0.0181	0.675	8
GBDT	0.0008	0.0282	0.0185	0.69	9
XGBoost	0.0008	0.0282	0.0185	0.6898	10
RF	0.0009	0.0292	0.0185	0.667	11
Knn	0.0011	0.0334	0.0199	0.5647	12

0.91, indicating a high level of accuracy in the predictions made by GRU for the seasonal series.

3) **Predictions of Remaining Components.**

The prediction of the remaining component, which represents the irregular fluctuation, was carried out by comparing RF with several other models. The results of the prediction were analyzed and tabulated in Table 5. Due to the fluctuation of the remaining component around 0, the MAPE criteria was not considered in this case, similarly to Table 4.

As evidenced by the results presented in Table 5, the efficacy of the prediction models in predicting the residual component of the egg price series is limited. The RF model emerged as the most effective, followed by BP and Adaboost. These results suggest that the ensemble learning class of ML methods is better suited for residual component prediction. When comparing the RF model with the second-best model, it is noted that the average MSE, RMSE and MAE of the RF model are reduced by 4.55%, 2.29% and 2.45%, respectively.

The residual component has a lower prediction accuracy compared to the trend and seasonal components. This can be attributed to the irregular fluctuations in the residual component, which are the result of a combination of external factors, such as epidemics, policies, and disasters, that are challenging to predict.

4) **Final Forecast Results.**

In order to assess the accuracy of the forecasting performance, a comparison was conducted between our proposed hybrid model and several other single and hybrid models. Considering the different data sets used, the methods of the existing literature were applied to the present data for comparison. Figure 5 displays the original price

Table 5. Statistical results of the predictions for the remaining component

Model	**MSE**	**RMSE**	**MAE**	R^2	**Rank**
RF	0.0015	0.0383	0.0299	0.4867	1
BP	0.0015	0.0392	0.0306	0.4625	2
Adaboost	0.0016	0.0394	0.031	0.4573	3
GRU	0.0016	0.0399	0.0296	0.444	4
LSTM	0.0016	0.0401	0.0318	0.4383	5
LightGBM	0.0017	0.0408	0.0323	0.4193	6
GBDT	0.0017	0.0417	0.0339	0.3939	7
TCN	0.0018	0.0422	0.0319	0.3787	8
RNN	0.0018	0.0426	0.0322	0.3675	9
Knn	0.0019	0.0432	0.0342	0.3487	10
XGBoost	0.0019	0.0437	0.0343	0.3331	11
SVR	0.0022	0.0463	0.0368	0.2499	12

series, as well as the forecasts of the several models. The proposed model can be seen to fit well. Furthermore, the results of the predictions are presented in Table 6.

As indicated by Table 6, our proposed model demonstrates superiority over its counterparts across each performance metric. Among the single models evaluated, TCN achieved the best results with the lowest error. The application of the STL method

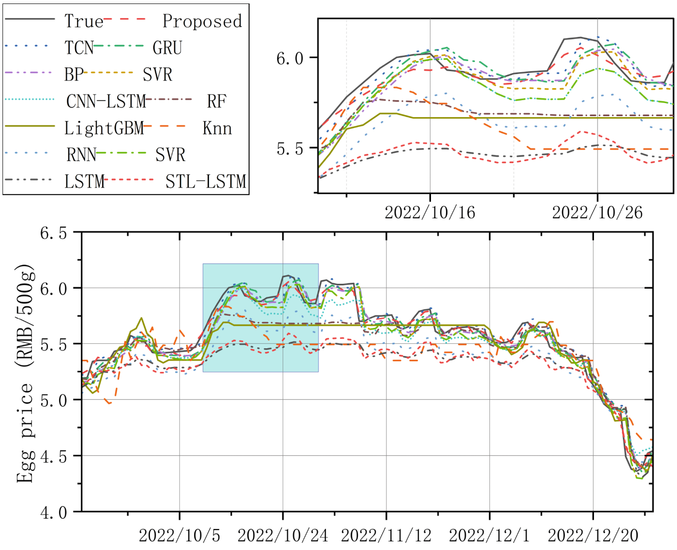


Fig. 5. Original price series and the predicted results of models.

Table 6. Statistical results of the predictions for egg price

Model	MSE	RMSE	MAE	MAPE	R^2	Rank
Proposed	0.0025	0.05	0.0414	0.7537	0.9831	1
STL-TCN-RF	0.0028	0.0525	0.0423	0.7731	0.9814	2
STL-TCN	0.0040	0.0633	0.0489	0.8877	0.9729	3
TCN	0.0074	0.0861	0.0592	1.0746	0.95	4
GRU	0.0087	0.0933	0.0698	1.2688	0.9412	5
Attention-TCN	0.009	0.0948	0.0708	1.2844	0.9393	6
BP[7]	0.0092	0.0957	0.0713	1.3035	0.9381	7
SVR	0.01	0.1000	0.0799	1.4592	0.9325	8
CNN-LSTM	0.0104	0.1019	0.0769	1.3950	0.9299	9
RF	0.0272	0.165	0.1191	2.1472	0.8162	10
LightGBM	0.0342	0.1849	0.1407	2.5551	0.7691	11
GBDT	0.0424	0.2058	0.1586	2.8745	0.7141	12
Adaboost	0.0479	0.2189	0.1659	3.0159	0.6765	13
RNN	0.0481	0.2193	0.1998	3.6767	0.6752	14
ARIMA [4]	0.0582	0.2412	0.1885	3.4542	0.6073	15
Knn	0.0779	0.279	0.2257	4.1855	0.4744	16
XGBoost	0.0859	0.2931	0.2102	3.8904	0.4199	17
STL-LSTM [12]	0.0899	0.2998	0.2606	4.8307	0.3932	18
LSTM [11]	0.0978	0.3127	0.2681	4.9818	0.34	19

on TCN (STL-TCN) resulted in a reduction of the MSE, RMSE, MAE, and MAPE by 45.41%, 26.48%, 17.48% and 17.39% respectively, compared to the single TCN model. The hybrid model, STL-TCN-RF, which replaced the TCN's prediction on the residual components with RF, further reduced the MSE, RMSE, MAE, and MAPE by 31.42%, 17.14%, 13.37%, and 12.91% respectively, compared to the STL-TCN model, demonstrating the effectiveness of RF in predicting the residual components. Finally, the addition of the GRU model to the hybrid model, STL-TCN-GRU-RF, resulted in a further reduction of MSE, RMSE, MAE, and MAPE by 9.54%, 4.69%, 2.2%, and 2.5% respectively, compared to STL-TCN-RF, indicating the superiority of GRU in predicting the seasonal component. These results demonstrate the good adaptability of TCN in egg price prediction, and the improvement in prediction accuracy achieved by our proposed hybrid model, which outperformed previous models. This paper investigates models with the addition of an attention mechanism (Attention-TCN), but with reduced prediction accuracy, suggesting that the attention mechanism is not well adapted to the data forecasting task. The analysis concluded that for the price forecasting task in this paper, the introduction of the attention mechanism caused the model to assign larger weights to individual time steps, which interfered with the overall trend of price changes

Table 7. Prediction accuracy measures of different time-step

Matric	**Time-step**					
	1	***2***	***3***	***4***	***5***	***6***
MSE	0.0019	0.0074	0.0108	0.0131	0.0253	0.0281
RMSE	0.0434	0.0864	0.104	0.1145	0.159	0.1675
MAE	0.0346	0.0645	0.0766	0.0878	0.1194	0.1305
MAPE	0.626	1.1953	1.42	1.6178	2.1852	2.4109
R^2	0.9873	0.9496	0.927	0.9115	0.8294	0.8106

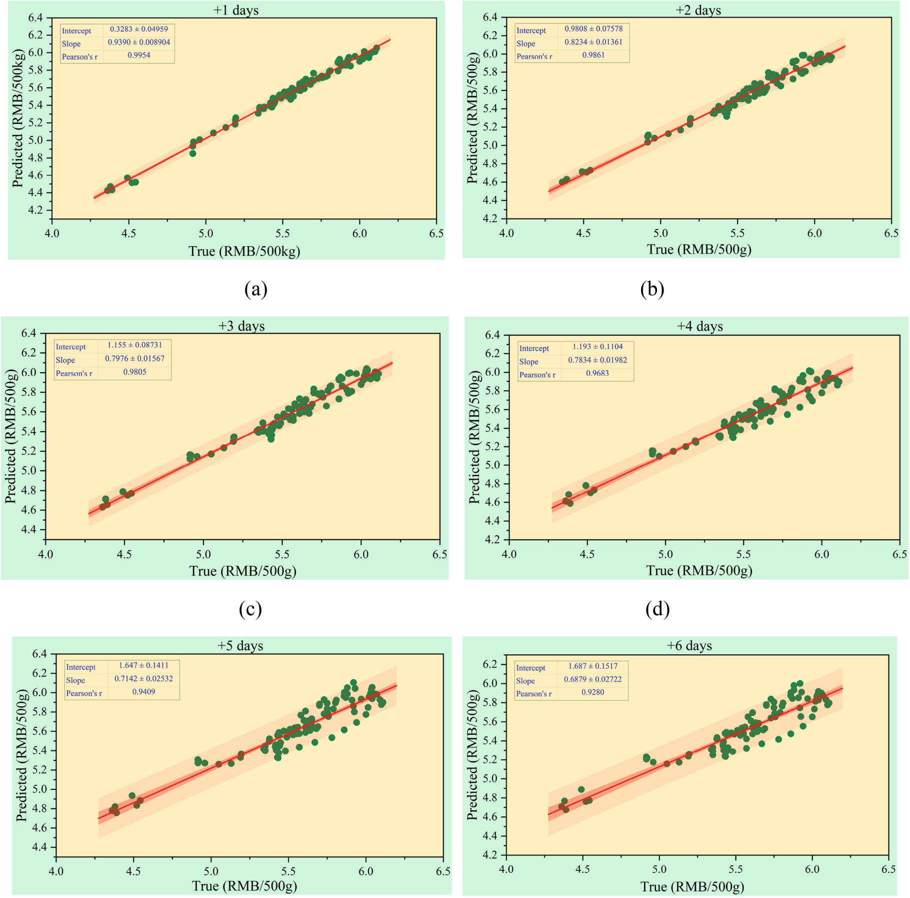
over time, resulting in a reduction in the model's forecasting accuracy. The TCN model has sufficient capacity to analyse the data, and the attention mechanism introduced does not take advantage of the processing of long time series data, but increases the training difficulty of the model, which ultimately reduces the prediction accuracy of the model.

5) **Predictions at Different Time Steps.**

Table 7 shows the performance measurement of the proposed model when the time-step is set to L $\in$ {1, 2, 3, 4, 5, 6}. The experimental results show that the prediction error gets larger as the prediction step size increases.

Figure 6 is a scatter diagram of eggs prices predicted by the proposed model for the next 1–6 days. Figure 6(a)–(f) are scatter diagrams of eggs price forecasts for the next 1 day, 2 days, 3 days, 4 days, 5 days, and 6 days, respectively. As the forecast step size increases, the points in the scatter plot gradually depart from the center line, indicating that the difference between the predicted and actual values grows. The similarity coefficient R^2 of proposed's prediction results on the six day is more significant than 0.8, so the prediction results on the six day can be generally considered to be reliable.

In summary, deep learning models have demonstrated their efficacy in predicting trend and seasonal components of time series data, while ensemble learning has proven to be effective in forecasting the residual component. Based on the evaluation of several models, it was concluded that TCN, GRU, and RF are the best performers in predicting the trend, seasonal, and residual components, respectively. As such, a hybrid model combining these algorithms was selected for the purpose of forecasting egg prices. This hybrid model leverages the strengths of each individual model to provide a more robust and accurate forecast. Compared to previous work, this improvement is attributed to the utilization of multiple models in multiple experiments, followed by the selection of the model with the best predictions for each component and their subsequent combination, resulting in a reduction in error.



(a) (b)

(c) (d)

(e) (f)

Fig. 6. Scatterplot of multi-step prediction of proposed model, (a) +1 days, (b) +2 days, (c) + 3 days, (d) +4 days, (e) +5 days, (f) +6 days.

4 Conclusions

This study proposes a novel STL-based hybrid model for forecasting the price of egg in China. First, the egg-price series is decomposed by STL into three additive components. Second, the resulting trend, seasonal and remaining components are experimentally selected for forecasting by the TCN, GRU and RF methods, respectively, and then the forecasts are aggregated. Finally, the forecasting performances of our models and the selected counterparts are compared and discussed. The empirical results show that our method effectively improves the prediction accuracy on the egg price series, and the proposed hybrid model outperforms the selected competitors. Meanwhile, the results of this study also provide new ideas and methods for price forecasting techniques for other agricultural products, which have some application value.

References

1. Yan Zhenyu, Sun Yangxue, Analysis of the fluctuation law and influencing factors of egg price in China. Statistics and Decision, 34: pp. 150–154, 2018.
2. Huamin Zhu, Ru Xu, Hongyao Deng, A novel STL-based hybrid model for forecasting hog price in China. Computers and Electronics in Agriculture, 198, 2022.
3. Yajiao Tang, Zhenyu Song, Yulin Zhu et al., A survey on machine learning models for financial time series forecasting. Neurocomputing, 512, pp. 363-380, 2022.
4. Zhang Yifan, Fan Meihua, Analysis and forecast of egg prices in national market based on ARIMA model, China Poultr, 42, pp. 82–86, 2020.
5. Du Xiya, Li Keqiang, Li Yuxiang, Prediction and analysis of egg prices in Hebei Province based on ARIMA model–Tantao Jinfeng wholesale poultry and egg farmers' market as an example, Modern Agriculture, vol. 11, pp. 39–41, 2021.
6. Gao Yang, An S. B, Comparative study on the effect of egg price forecasting in China–a comparative analysis based on BP neural network model and egg futures forecasting model, Price Theory and Practice, vol. 3, pp. 75–78, 2021.
7. Liu LJ,Lou WG, Application of GRNN model in egg price forecasting. China collective economy, vol. 36, pp. 66–67, 2014.
8. Zhou Rongzhu, Liu Heguang, Analysis of egg price prediction based on grey model and neural network model, Chinese Society of Animal Husbandry and Veterinary Medicine, vol. 402, 2016.
9. Liu Xue, Liu Jintao, Li Jiali, Zhang Xiaobang, Zhang Wenhao, Egg price forecasting in Beijing based on seasonal decomposition and long short-term memory, Journal of Agricultural Engineering, vol. 36, pp. 331–340, 2020.
10. Yan W, Short-term egg price forecasting based on EEMD-LSTM combined model, Nanjing Agricultural University, 2018.
11. Yin, H., Jin, D., Gu, Y.H., Park, C.J., Han, S.K., Yoo, S.J, Vegetable Price Forecasting Using STL and Attention Mechanism-Based LSTM. Agriculture, vol. 10(612), 2020.
12. Xiong Tao, Li, Chongguang and Bao Yukun, Seasonal forecasting of agricultural commodity price using a hybrid STL and ELM method: Evidence from the vegetable market in China, NEUROCOMPUTING, vol. 275, pp. 2831–2844, 2018.
13. Cleveland R, Cleveland W, McRae J, et al, STL: A seasonal-trend decomposition procedure based on loess, Journal of Official Statistics, vol. 6, pp. 3–33, 1990.
14. Bai s, Kolter J Z, Koltun V, An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, arXiv, 2018.
15. Pei Y, Huang C.-J., Shen Y., Wang, M., A Novel Model for Spot Price Forecast of Natural Gas Based on Temporal Convolutional Network, Energies, vol. 16, pp. 2321, 2023.
16. Wang Zepeng, Chen Xiaoyan, Pang Tao, Yu Fu, Hu Xiaonan, Wang Zhen, An improved time-convolutional network-based pig price forecasting method, Journal of China Agricultural University, vol. 26(12), pp. 137–144, 2021.
17. HE K, ZHANG X, REN S, et al, Deep residual learning for image recognition, Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778, 2016.
18. Chung J, Gulcehre C, Cho K, et al, Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling, arXiv: Neural and Evolutionary Computing, 2014.
19. Reiman L., Random forests, Machine Learning, vol. 45, pp. 5–32, 2001.
20. Big Data for Agriculture and Rural Areas, https://www.agdata.cn.