Deep Learning Based Customer Preferences Analysis in Industry 4.0 Environment

Qindong Sun[1,2] · Xingyu Feng[2] · Shanshan Zhao[3] · Han Cao[2] · Shancang Li[3] · Yufeng Yao[3]

Abstract

Customer preferences analysis and modelling using deep learning in edge computing environment are critical to enhance customer relationship management that focus on a dynamically changing market place. Existing forecasting methods work well with often seen and linear demand patterns but become less accurate with intermittent demands in the catering industry. In this paper, we introduce a throughput deep learning model for both short-term and long-term demands forecasting aimed at allowing catering businesses to be highly efficient and avoid wastage. Moreover, detailed data collected from a business online booking system in the past three years have been used to train and verify the proposed model. Meanwhile, we carefully analyzed the seasonal conditions as well as past local or national events (event analysis) that could have had critical impact on the sales. The results are compared with the best performing forecast methods *Xgboost* and autoregressive moving average model (ARMA), and they suggest that the proposed method significantly improves demand forecasting accuracy (up to 80%) for dishes demand along with reduction in associated costs and labor allocation.

Keywords Artificial intelligence · Time series analysis · Sales forecasting · Long short-term memory (LSTM)

1 Introduction

In the past few years, *Big data analytics* and *deep learning* have been successfully applied in a number of applications, such as computer vision, artificial intelligence, natural language processing, *etc.*, and they now become important assets in business intelligence [1]. In the ever changing catering industry, big data analytics has been widely used to improve personalized marketing, customize services, analyze customers' eating habits and dining patterns, *etc.* [2, 3].

Deep learning-based demand forecasting is driving changes in the catering industry, not only because it is a powerful way of making contributions towards reducing costs and gaining competitive advantage, but also it delivers an intelligent and sustainable future [1–3]. Using historic data, it is possible for example to predict when and what ingredients will be needed during a certain period of time; this helps to anticipate the deliveries, which results in a more efficient system. In the past few years, a number of sales forecasting methods have been developed, including linear regression, exponential smoothing [2], the *Autoregressive Moving Average model (ARMA)* and so on. These models can accurately predict linear sequence, but they are unable to perform non-linear sequence prediction. Nowadays deep learning (DL) can be used for both linear sequence and non-linear sequence prediction for speech recognition [4, 5], image processing, and other artificial intelligence tasks [6].

This paper aims to develop an efficient demand forecasting model that can accurately predict the dishes

✉ Shanshan Zhao
    Shanshan.Zhao@uwe.ac.uk

Qindong Sun
qdongsun@xjtu.edu.cn

Xingyu Feng
fengxingyu@stu.xaut.edu.cn

Han Cao
caohan@stu.xaut.edu.cn

[1] School of Cyber Science and Engineering, Xi'an Jiaotong University, 710049, Xi'an, China

[2] Shaanxi Key Laboratory of Network Computing and Security, Xi'an University of Technology, 710048, Xi'an, China

[3] University of the West of England, Bristol, UK

sales both in short- and long-term with lower prediction error. The main contributions are summarized as follows:

1. Using deep learning, a real time sales demand forecasting model is proposed that can accurately predict demand based on past trends;
2. Both accurate short- and long-term predictions will be very helpful in developing an efficient delivery system for supplies;
3. The trained model can achieve prediction rate accuracy higher than 80%, outperforming existing methods such as *Xgboost (less than 60%)* and *ARMA*.

2 Related work

With the rapid development of the Internet and information technology, the available operational data in the commercial field have been drastically increased [1]. The emergence of business intelligence can transform the enterprises' operation data to an asset with high commercial value. Enterprises are now able to accumulate potential information from massive data and, therefore, optimize their system and improve customer satisfaction. Thus, business intelligence can provide better service for the customer and create greater profits for the enterprises.

Sales forecasting involves methodologies that predict the sales volume of commodities during specific periods in the future, taking into account the historical sales of commodities. The accurate prediction of future short-term sales of goods is vital to enterprises. The time series model is one of the most common models for sales forecasting. The ARMA is a linear model for predicting time series. Ramos et al. investigated the performance of *ARIMA* and state space model for retail sales forecasting [7]. Sales prediction is not only related to historical sales data, but it can also be affected by external factors. Arunraj et al. proposed a Seasonal Autoregressive Integrated Moving Average with external variables (*SARIMAX*) model [8], which improves the traditional Seasonal Autoregressive Integrated Moving Average (*SARIMA*) [9].

Machine learning based analytical methods have been widely used in the study of sales forecasting. Wen et al. applied support vector machines to predict grape sales in a fruit shop [10]. Machine learning techniques was used to forecast sales of a drug store company by Gurnani et al. in [11]. Holmberg M and Halldén P forecast the restaurant sales with Xgboost and LSTM Neural Network [12]. Researchers have also applied neural networks to sales forecasting. The BP neural network was exploited to forecast construction project cost [13]. Abedinia et al. proposed a hybrid forecasting approach based on combination of a neural network with a metaheuristic algorithm

to predict solar power [14]. The development of artificial neural networks and the continuous improvement of computers' processing power have prompted the rise of deep learning [15]. In the continuous development of deep learning technology, a variety of neural network structures have emerged for the study of time series, such as recurrent neural network (RNN) [16] and its variation LSTM [17]. Boné and Assaad also used RNN in the study of time series forecasting [18]. Kaneko and Yada proposed a model that can predict adequately retail sales of merchandise using deep learning [19]. LSTM has also been used for investigations on time series problems such as short-term traffic prediction [20], power load forecasting [21], and forecast dynamic destination-destination (OD) matrices in a subway network, the model structure contains one LSTM layer, the input attributes of the LSTM models are OD counts of the 300 timestamps preceeding the timestamp to forecast dynamic OD matrices. [22], *etc.*

In the catering industry sales forecasting, the collected data cover a wide range of dishes, including linear sales data and non-linear sales data. And forecast models such as ARMA and SARMA do not represent nonlinear data well, hence the sales forecast accuracy drops during the holidays. In comparison, RNN and LSTM perform better than BP prediction models to a small extent in time series data. This motivated us to use deep learning to forecast the dishes sales aiming not only to circumvent the shortage of the traditional sales forecasting methods, but also to excavate efficiently the sales' trend characteristics obtaining better forecasting results.

In deep learning, the recurrent neural network (RNN) is widely employed to deal with time series related prediction. Actually, gradient explosion and gradient disappearance in long-term memory training is a big problem. Hochreiter and Schmidhuber in 1997 proposed the Long Short-Term Memory network (LSTM) which improves the traditional RNN [16]. The LSTM network adds a cell state c for preserving long-term states and adopts the concept of gate:

– **Forget gate**, is used to control the update of the cell states and determines how the cell state c_{t-1} at time $t-1$ is kept until cell state c_t at time t.
– **Input gate**, determines the current network input x_t saved to the content of cell state c_t.
– **Output gate**, can control how much of the current cell status is filtered out.

Figure 1 shows the structure of the memory unit of the LSTM model. In Fig. 1, f_t, g_t and O_t represent the *forget gate, input gate* and *output gate*, respectively. c_{t-1} and h_{t-1} represent the *cell state* and *output values* at time $t-1$. And c_t, h_t correspond to the values at the time t. σ and $tanh$ represent sigmoid and tanh functions. In the last few years, a few variants of LSTM have been developed, including

Fig. 1 Structure of the memory unit of LSTM
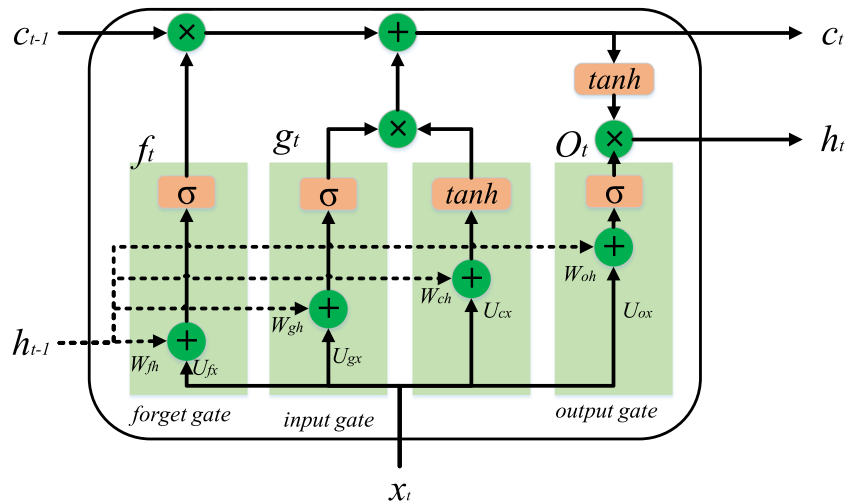
"Peephole connections" [23] and Gated Recurrent Unit (GRU) [24]. The *Peephole connections* is an extending LSTM with "Peephole Connections" and the GRU is used in the modeling of Sequence [25]. Chung et al. proved the superiority of the gating unit by evaluating several traditional sequence models including RNN, LSTM and GRU [25].

3 Proposed method

This section proposes a short-term prediction model which includes the following four steps: (1) A pre-processing step is introduced before classifying the original sequence into *ordinary working days data* and *holiday data*; (2) Sales forecast models for both *ordinary working days data* and *holiday data* are built; (3) Historical data is used as input to further train the models; (4) According to the forecast date, select the appropriate model for prediction; Fig. 2 shows the general architecture of the proposed forecast model.

3.1 Data partitioning based on date characteristics

In order to train the model we use a dataset that contains billing information acquired from a hotpot chain (duration

two and half years). It should be noted that in the pre-processing state the original data contain missing values for some random time periods. Thus, when using the original data sequence for direct prediction, the incomplete sequence cannot be applied. In traditional prediction methods, missing data are usually filled with an artificial mean or median and the traditional prediction models treat *ordinary working days data* and *holiday data* in the same way; this causes a low forecast accuracy. To improve the forecast accuracy, we divide the raw data into two parts based on the *date characteristics*: *ordinary working days data* and *holiday data*. The missing data are discarded to avoid the influence of data noise.

We noticed that the dish sales data follow a sinusoidal distribution, and the period is seven working days. Sales data on the same weekday have similar features: the largest sales occurred at the period during the weekend, while the curve from the weekend to the upcoming Tuesday shows a downward trend and then an upward trend. Therefore, the original data are divided into seven time sequences: Monday sales data sequence to weekend sales data sequence.

During the statutory holidays (e.g. in China, *National Day, Mid-Autumn Festival, etc.*) or some non-statutory holidays (e.g., *Mother's Day, Valentine's Day, etc.*), most people prefer eating out at restaurants. This causes an

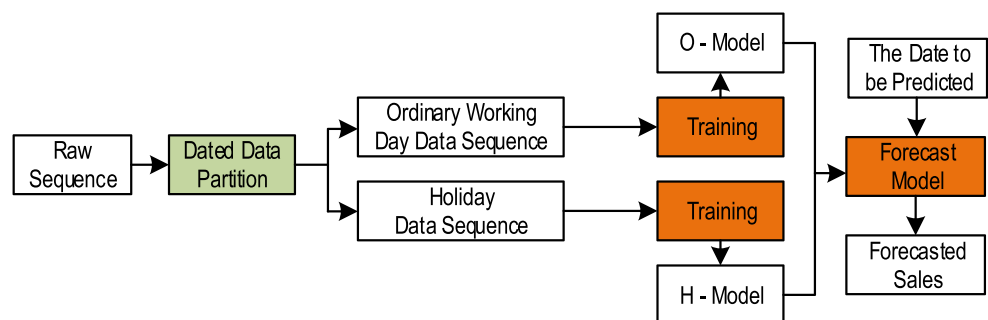**Fig. 2** Framework diagram of the forecasting model

Table 1 Holiday information in China

Holiday	Days	Date	Name
Statutory Holiday	3	1st Jan	New Year's Day
		5th April	Tomb-Sweeping Day
		1st May	Labour's Day
		5th Lunar May	Dragon-boat Day
		15th Lunar Aug	Mid-Autumn Festival
	7	1st Oct	Chinese National's Day
	7	1st Lunar Jan	Chinese New Year's Day
Non-Statutory	1	14th Feb	Valentine's Day
		8th Mar	Women's Day
		2nd Sunday May	Mother's Day
		3th Sunday Jun	Father's Day
		7th Lunar Jul	Double-seventh Day
		25th Dec	Christmas
		15th Lunar Jan	Lantern Festival

increase of people's flow and further leads to a larger volume of the dishes' sales when compared to the previous sales of the same weekday. Separating the days into *holiday* or *ordinary day* we can significantly improve the forecast accuracy of dishes sales. As an example, 14 holidays in China associated with sales of dishes are listed in Table 1.

Figure 3 address the data partitioning, in which the data is categorized into two groups: sequences of ordinary working day, and sequences of holidays. The sequences of ordinary working day are categorized into each day and the sequences of holidays are categorized into sequences on the New Year's Day and the Tomb-sweeping Day (Tomb-sweeping Festival is a major festival of spring sacrifice in China, generally with three days' holiday) for further analysis.

As shown in Fig. 3, each sub-figure represent results on different working days, e.g., 2015/01/01, 2015/01/08,
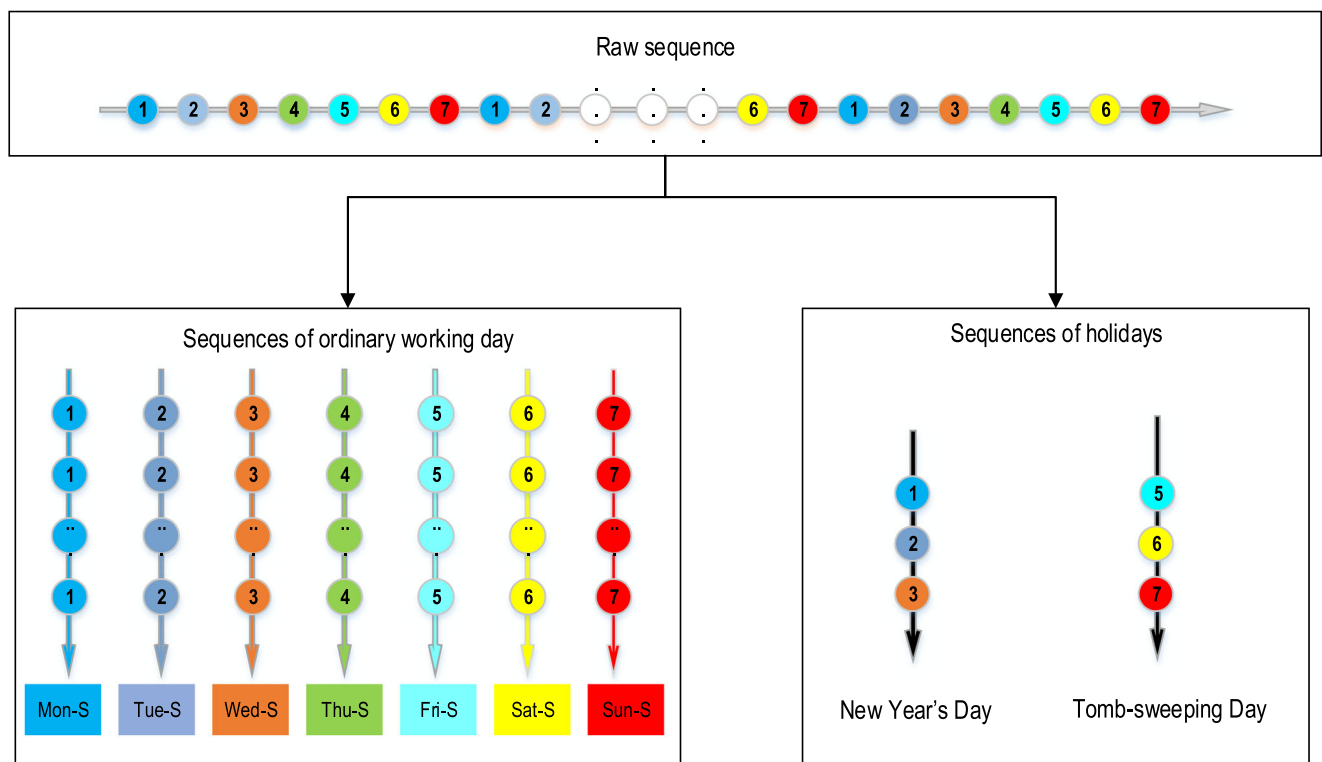

Fig. 3 Data partitioning

2015/01/15 are all `Thursday`, so the sales data of these dates will be divided into the Thursday sequence. Sales data for these dates such as 2015/01/02, 2015/01/09, 2015/01/16 are divided into `Friday sequence`, and so on. If it belongs to a holiday, it is divided into corresponding data sequences according to the date.

3.2 Detection and processing of outliers

Outliers refer to individual values that deviate significantly from the rest of the observations of the samples they belong to. Sales of dishes can be affected by external factors. Therefore there exist some abnormal data-points which are obviously higher than or below the normal value, and their values exceed the fluctuation range. The existence of outliers will influence forecasting precision to some extent. Accordingly, the abnormal data points should be removed in the training process of the model.

We identify abnormal data points in this paper by using the single variable outlier detection method based on the Gaussian distribution. When the data point falls outside the

interval: $R : [(\bar{x} - 2\sigma), (\bar{x} + 2\sigma)]$, we say it is an outlier. Note that

$$\bar{x} = \frac{1}{k} \sum_{i=1}^{k} x_i \tag{1}$$

$$\sigma^2 = \frac{1}{k} \sum_{i=1}^{k} (\bar{x} - x_i)^2 \tag{2}$$

where $\bar{x}$ denotes sample mean and σ^2 is sample variance.

3.3 Sales forecast model

Our sales model includes a forecast model of ordinary working days and a forecast model of holiday sales, as shown in Fig. 4. In this paper we refer to the ordinary working day prediction model as the *O - Model*, and the holiday prediction model is named *H - Model*. The sales data of ordinary working days are seven single variable sequences. We use the LSTM network to model them, and the model structure is shown on the left of Fig. 4. The *O - Model* contains an input layer, a hidden layer and an output
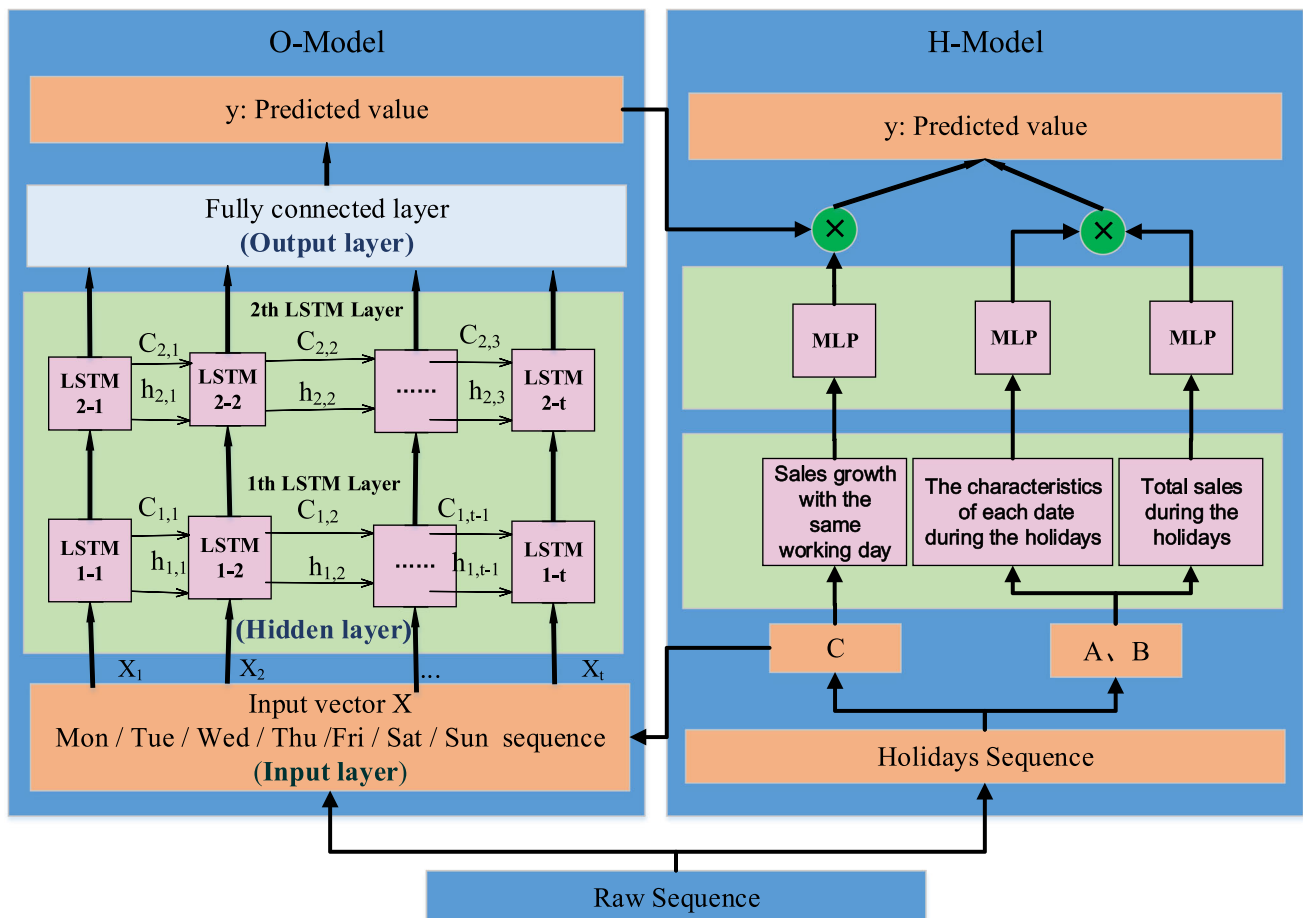

Fig. 4 Proposed Forecast Model

layer. The hidden layer is composed of two LSTM layers and their number of neurons is 56 and 128, respectively; they all use *tanh* as the activation function. The output layer is fully connected, using sigmoid as the activation function. In Fig. 4, X_i represents the input vector at time i, $c_{i,j}$ represents the j-th cell state of the i-th LSTM layer, and $h_{i,j}$ represents the output of the the j-th neuron of the i-th LSTM layer. Moreover, y is the prediction value. The $O - Model$ input is the historical sales data of a workday, and the output is the sales forecast of the next same working day.

Holiday sales data deviations are not usually the same for distinct periods. For example, the duration of the *Mid-Autumn Festival* and the *Tomb-sweeping Day* is three days off, as they considered to be traditional Chinese festivals. During the *Mid-Autumn Festival*, more people usually go out to eat on the exact day of the *15th* day of the eighth month of lunar calendar. Thus, the day's sales volume of dish products account for a large proportion of total holiday sales. While during the *Tomb-sweeping Day*, when people go out to eat, they do not care about the time they will be eating out. Therefore, it is necessary to mine the characteristics of sales data during different holidays.

According to the holiday statistics, after carrying out a detailed analysis of each holiday, we divide the holidays into three categories on the basis of the days of holiday:

– **A**: *New Year's Day, Tomb-sweeping Day, International Labour Day, The Dragon Boat Festival, Mid-Autumn Festival*;
– **B**: *Spring Festival* and *Chinese National Day*;
– **C**: *Non-statutory holiday*.

In spite that there might exist possible overlap of holidays near the time node, or limited amount of data and single day holiday sales fluctuations, we noted that the total sales volume of dishes during the same holidays was relatively stable. Therefore, in order to forecast the **A, B** types of holiday sales of food we perform two actions. First, we predict the total sales of dishes during holidays. Second, based on historical holiday sales data for each day of the holiday period, we assign weights to get the holiday daily sales forecast.

However, insufficient data volume makes it difficult for us to predict the total sales during the holidays with the existing models. We came up with the idea to predict them using the relationship between the data on the eve of the holidays and the total sales during the holidays. The model's input is the sales of dishes on the eve of holidays and the output is the total sales of dishes on holidays as a percentage of the inputs. The Multi-layer Perceptron (MLP) includes one input layer containing 7 neurons, two hidden layers, and the output layer contains a neuron. When the total sales volume of a holiday has multiple forecasts, we then average these forecasts as the dish sales of every day during the holidays.

For **C** holidays, there are also two steps to predict sales volume. The first step is to perceive the **C** type of holiday as a normal business day sales forecast. The second step is to predict sales growth based on historical holiday data and the same working day data, and use them together to obtain the final prediction value.

The first part of the predictions in the **A, B**, and **C** type holidays forecast in H - *Model* are all recorded as P_o. Two data sequences were extracted from **A** and **B** type holidays for the aims of training. The weight coefficient is calculated by MLP. The input and output layers contain N neurons, where N represents the days of holidays. The MLP includes two hidden layers. Assuming that the weight coefficient is $(\omega_1, \omega_2, \ldots, \omega_N)$, the predicted value of sales every single day in the holiday is:

$$(p_1, p_2, \ldots, p_N) = P_o(\omega_1, \omega_2, \ldots, \omega_N) \tag{3}$$

The sales growth forecast in type **C** is predicted by an MLP. The model structure is: $4, 6, 12, 4$. The sales volume of dishes on the day of the holidays is S_h, the input vector is the historical sales values S on the same business day as the holidays, and the output vector α can indicate the growth of sales, $(S_h - S)$, accounted for the percentage of sales S in sales volume. The predicted value is:

$$P = P_o + (S \cdot \alpha) \tag{4}$$

3.4 Training Algorithm

The O - *Model* uses the LSTM network and the training algorithm is the back-propagation algorithm with four steps:

Step 1: Calculate the forward output value. The model input uses the Min-Max normalization method to normalize the data of the ordinary working day sequence, then forwardly calculate the output value according to the following equations:

$$f_t = \sigma(W_{fh}h_{t-1} + U_{fx}x_t + b_f) \tag{5a}$$

$$g_t = \sigma(W_{gh}h_{t-1} + U_{gx}x_t + b_g) \tag{5b}$$

$$O_t = \sigma(W_{oh}h_{t-1} + U_{ox}x_t + b_o) \tag{5c}$$

$$c_t = f_tc_{t-1} + g_t \cdot tanh(W_{ch}h_{t-1} + U_{cx}x_t + b_c) \tag{5d}$$

$$h_t = tanh(c_t)O_t \tag{5e}$$

In the above formula, f_t, g_t, O_t, respectively, symbolize the **forget gate, input gate** and **output gate**. x_t denotes the input at time t, h_{t-1} denotes the output at time $t-1$, and W, U, b indicate the corresponding weight of h_{t-1}, the weight of current input x_t, and the bias term, respectively. σ represents a sigmoid activation function.

Step 2: Obtain the error term of each neuron in LSTM in the reverse direction using the mean square error as the error calculation formula. The error propagation direction of the LSTM network includes the time direction and the layer direction.

Step 3: Calculate the gradient of each weight according to the error term of the neural unit.

Step 4: Update the weight and bias through the optimization algorithm Adaptive Moment Estimation (Adam) [26].

Repeat the above steps until the training is over. MLP also use the back-propagation algorithm to train the model. The training steps are similar to those of the LSTM model, but the forward calculate output value and the reversely calculate error item are simpler. We will not repeat here.

4 Experimental results and analysis

The dataset used in this paper are based on a two and a half year's real billing data of a well-known catering business. The time span of data is from `January 1, 2015 to July 23, 2017`. We processed the original billing data and calculated the sales of each item in days. In this work, all 147 dishes are divided into ten categories based and labelled from $I - X$. In order to achieve the forecasting goal, we train the model using data from 2015 to 2016. The data of 2017 are used to validate the model. We evaluate the model by its prediction accuracy rate. The accuracy rate can be derived from Eq. 6

$$H = (1 - abs((P - T)/T) \times 100\% \qquad (6)$$

where H indicates the accuracy rate, P is the predicted value, T is the true value, and abs is the absolute value function. In order to evaluate the forecasting performance of each method, the mean absolute percentage error (MAPE) be used to compare forecasting accuracy of each model in this paper. The calculating equation is showed as

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} |\frac{\bar{y}_i - y_i}{y_i}| \qquad (7)$$

in which $\bar{y}_i$ and y_i denote the actual and predicted values, respectively. The hit rate R_h is the forecasting accuracy.

$$R_h = 1 - MAPE \qquad (8)$$

Based on the date characteristics, we divide the original sequence into a ordinary working day sequence and a holiday sequence. As discussed previously, we use different models to predict these sequences and we utilize the *ARMA* and *Xgboost* models to make comparative experiments.

4.1 Prediction accuracy for ordinary working day

Figure 5 shows the comparison between our model against ARMA and Xgboost's predicted sales of 10 dishes on ordinary working day which are randomly selected one from each type of dishes (categories I - X). The data points in the figure contain only ordinary working day data and do not include holiday data. It can be seen from the line chart that our model predicts sales more accurately, this is not only reflected in the 10 dishes in the picture. We have made statistics on the prediction results of all dishes of the three models in the ordinary working day, and found that our model performs better for most dishes, and the prediction results of 27 dishes are not ideal. However, our results are not very different from the best ones. The max difference is 9.31% (ID: 3010057).

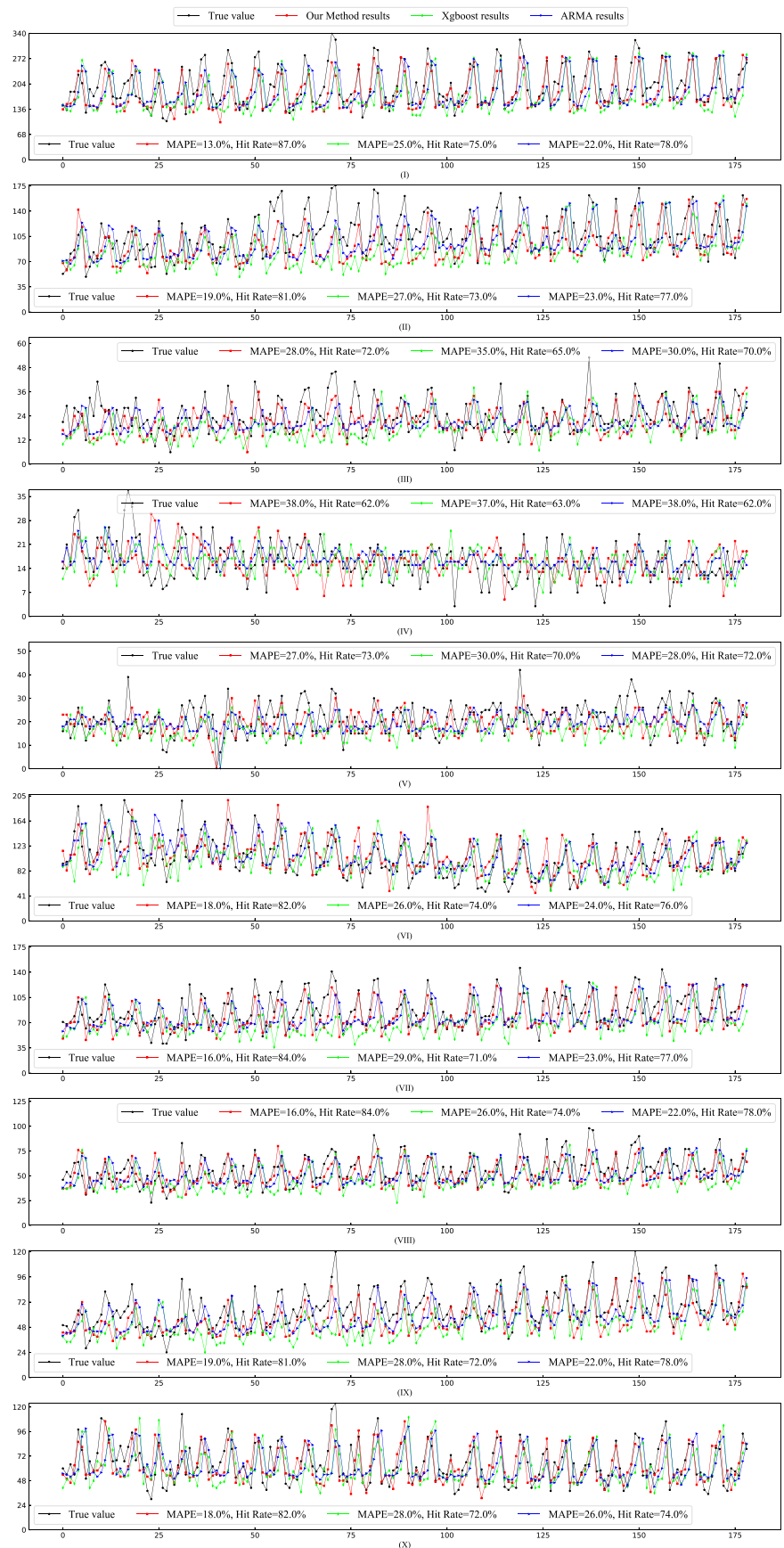### 4.2 Holiday forecast results

Holidays affect consumer behavior. During the holidays, an increase of people flow will lead to a rise in dish sales. If we use the traditional (i.e. ordinary days') prediction models to forecast the sales of dishes, we will inevitably encounter imprecise dishes sales forecast for the holidays. Figure 6 shows the predicted results for 10 randomly selected dishes from 147 dishes in categories **A** and **B**. Figure 6 demonstrates that dishes sales of holiday **A** and **B** have a clear upward trend compared with that of the same business day. Additionally, the forecast results derived from the *ARMA* and *Xgboost* methods are lower than the actual sales. For most dishes, the holiday sales forecast accuracy using our method is significantly higher than the predicted by the *ARMA* model and the *Xgboost* method. Nevertheless, there are still some deviations between the predicted and true values for some days. The max difference is 31.58% (ID: 3010011). So, the forecast is not ideal.

Figure 7 showcases the predicted results in the **C** type holidays category. Different from **A** and **B** holidays, not all sales volume in **C** holidays present an upward trend. Also, different dishes show different rising and falling trends on the same **C** type holiday. Therefore, the forecast accuracy for certain dishes in **C** type holiday failed to meet the requirements.

It can be seen that the sales forecasting accuracy is significantly improved after data partition occurred. We noted that the prediction for almost all dishes is accurate but there are still a few dishes having a forecasting rate less than 60%. This could be caused by the following reasons:

(1) *On individual dates, the actual sales of dishes are abnormal*. The sales of some dishes suddenly dropped to single digits, however the sales volume was normal before and after those dates. Among the 147 dishes,

Fig. 5 Comparison between predicted values and real values (The data in the figure shows the true and the predicted value of nine dishes in the remaining days removing the holidays in the time period, 2017/1/1 - 2017/7/23)
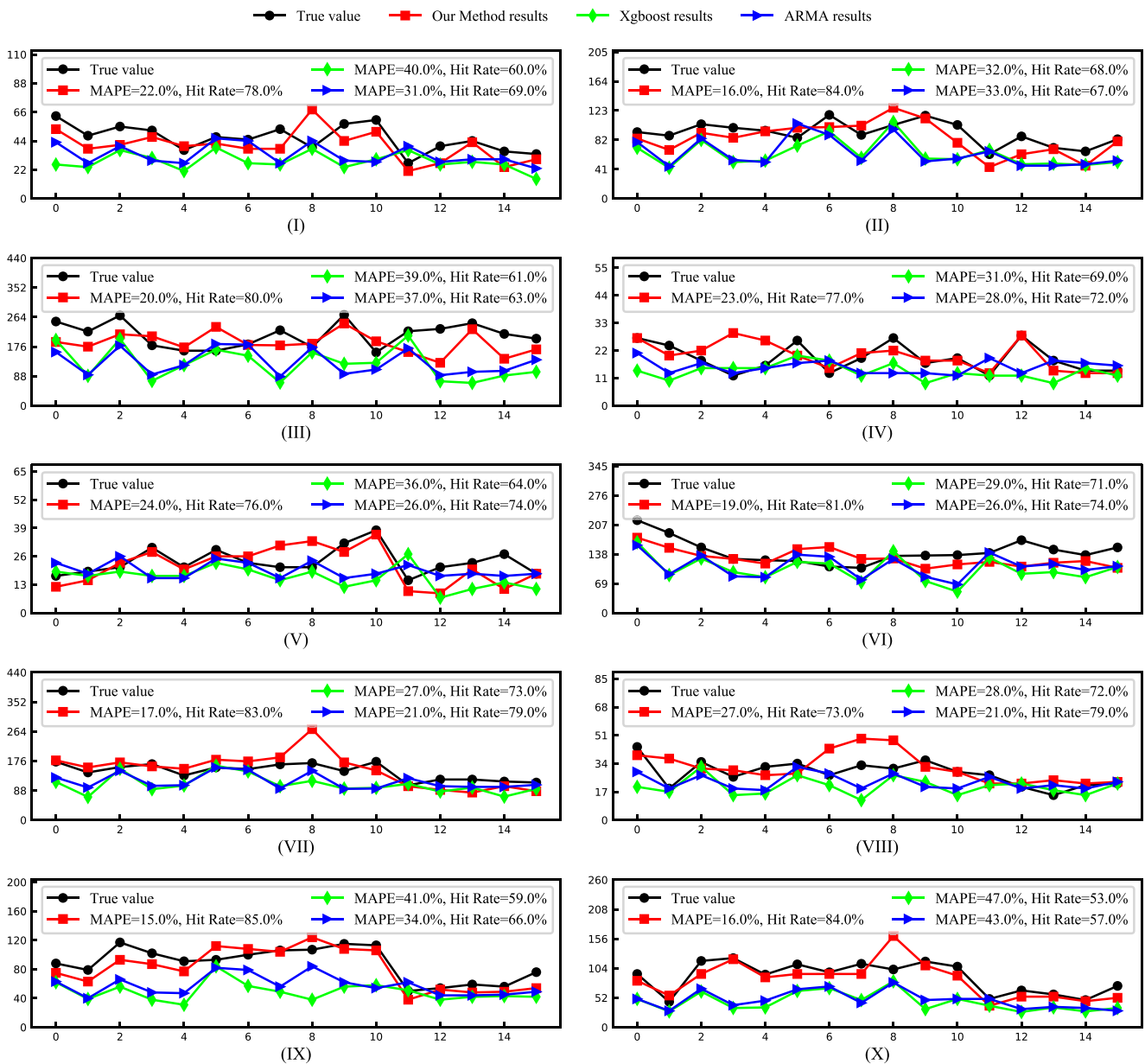
Fig. 6 The experimental results of **A** and **B** holidays

15 dishes have abnormalities in the real data. If we do consider such abnormal data when calculating the average of the sales forecasting accuracy rate, the mean forecasting accuracy rate of the 15 dishes can be more than 70%.

(2) *When the daily sales of dishes are low, the average accuracy rate decreases.* As shows in Fig. 5-7, in which when the daily sales of the dishes is low, the accuracy rate is relatively sensitive. This situation exists not only in dishes with accuracy less than 60%, but also in sales forecasting of other dishes. In dishes with an accuracy of less than 70%, there also exist dishes with less daily sales. Therefore, the forecast

accuracy rate of these dishes is lower, but considering that the absolute error is small, it is eventually within an acceptable range.

(3) *The individual peak data failed to fit well.* The sales of dishes have risen sharply in some days, resulting in our low forecast. But based on the available data we cannot speculate on the reasons of sales growth, as shown in Fig. 8.

Keras is used to build model. The computational complexity of the model is not high. The O-Model parameters are 110529 and the size is 1.3M. Predicting the sales of all dishes in a single day takes 1.02×147 ms on
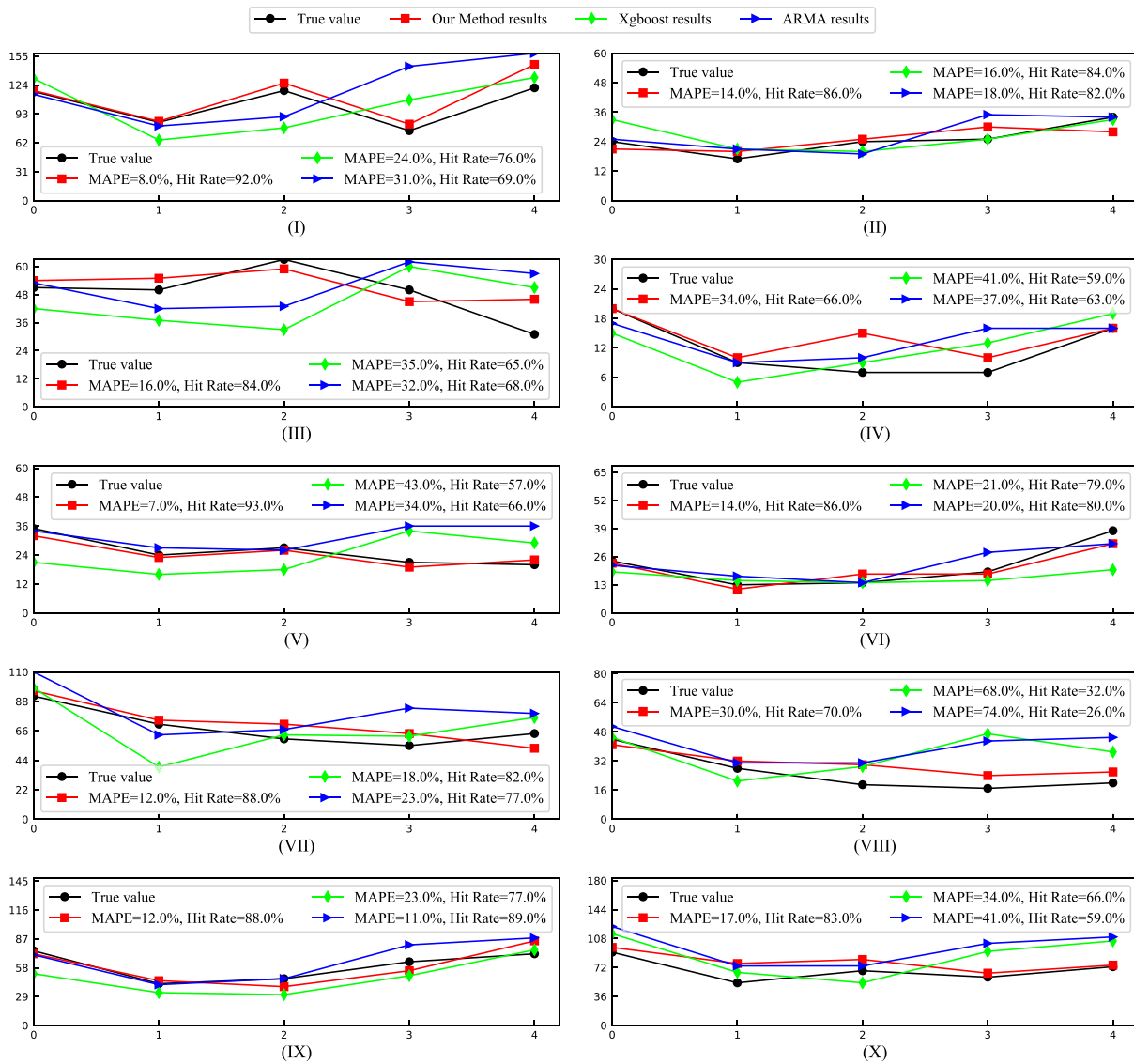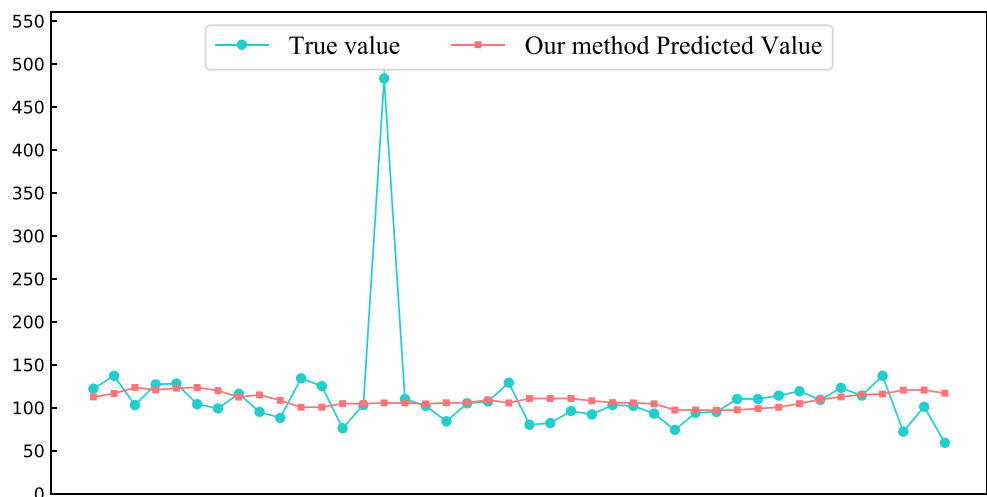
Fig. 7 The experimental results of **C** type holidays

Fig. 8 The individual peak data

Inter Core i5-8400 CPU. H-Model is less. So, it's advisable. The above experimental results show the superiority of the proposed method. Most of the experimental results have high prediction accuracy, but the prediction of some peaks and valleys failed to reach the expected results. After conducting a detailed analysis, we concluded that the reason for this situation is due to external factors on consumer behavior. Sales might have risen substantially because of product promotions, but until now we have not been able to get business promotion information.

5 Conclusion

Based on real data provided by a well-known restaurant, this paper constructs a sales forecasting model based on deep learning. We split the original time series into ordinary working day data and holiday data on the basis of date characteristics of the original sequence, and then we model them separately to form a complete forecast model of sales volume of dishes. The experimental results show that our model occupies preferable predication performance and stronger robustness, which can reflect the rules and characteristics of dish sales more comprehensively. The forecasting model can provide a powerful reference for the purchasing material of the purchaser, and has great potential in applications.

References

1. Chaudhuri S, Dayal U, Narasayya V (2011) An overview of business intelligence technology. Commun ACM 54(8):88–98
2. Brown RG, Meyer RF (1961) The Fundamental Theorem of Exponential Smoothing. Oper Res 9(5):673–687
3. Jin S, Zhang Z, Chakrabarty K, Gu X (2019) Anomaly detection and health-status analysis in a core router system. IEEE Design Test 36(5):7–17
4. Hinton G, Deng L, Yu D, Dahl GE, Mohamed A, Jaitly N et al (2012) Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Proc Magaz 29(6):82–97
5. Kolbæk M, Yu D, Tan Z, Jensen J (2017) Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks. IEEE/ACM Trans Audio Speech Language Proc 25(10):1901–1913
6. Gharehbaghi A, Lindén M (2018) A deep machine learning method for classifying cyclic time series of biological signals using time-growing neural network. IEEE Trans Neural Netw Learn Syst 29(9):4102–4115
7. Ramos P, Santos N, Rebelo R (2015) Performance of state space and ARIMA models for consumer retail sales forecasting. Robot Comput Integr Manuf 34:151–163
8. Arunraj NS, Ahrens D, Fernandes M (2016) Application of SARIMAX model to forecast daily sales in food retail industry. Int J Operat Res Inf Syst 7(2):1–21
9. Arunraj NS, Ahrens D (2015) A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting. Int J Prod Econ 170:321–335
10. Wen Q, Mu W, Sun L, Hua S, Zhou Z (2014) Daily sales forecasting for grapes by support vector machine. In: Li D, Chen Y (eds) Computer and computing technologies in agriculture VII. Springer, Berlin, pp 351–360
11. Gurnani M, Korke Y, Shah P, Udmale S, Sambhe V, Bhirud S (2017) Forecasting of sales by using fusion of machine learning techniques. In: 2017 International conference on data management, analytics and innovation (ICDMAI). IEEE, pp 93–101
12. Holmberg M, Halldén P (2018) Machine learning for restaurant sales forecast
13. Wang X (2018) Forecasting construction project cost based on BP neural network. In: 2018 10th International conference on measuring technology and mechatronics automation (ICMTMA). IEEE, pp 420–423
14. Abedinia O, Amjady N, Ghadimi N (2018) Solar energy forecasting based on hybrid neural network and improved metaheuristic algorithm. Computat Intell 34(1):241–260
15. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444
16. Elman JL (1990) Finding structure in time. Cogn Sci 14(2):179–211
17. Greff K, Srivastava RK, Koutník J, Steunebrink BR, Schmidhuber J (2016) LSTM: A search space odyssey. IEEE Trans Neural Netw Learn Syst 28(10):2222–2232
18. Boné R, Assaad M, Crucianu M (2003) Boosting recurrent neural networks for time series prediction. In: Pearson DW, Steele NC, Albrecht RF (eds) Artificial neural nets and genetic algorithms. Springer Vienna, Vienna, pp 18–22
19. Kaneko Y, Yada K (2016) A deep learning approach for the prediction of retail store sales. In: 2016 IEEE 16th International conference on data mining workshops (ICDMW), pp 531–537
20. Zhao Z, Chen W, Wu X, Chen PCY, Liu J (2017) LSTM network: a deep learning approach for short-term traffic forecast. IET Intell Transp Syst 11(2):68–75
21. Zheng H, Yuan J, Chen L. (2017) Short-term load forecasting using EMD-LSTM neural networks with a Xgboost algorithm for feature importance evaluation. Energies 10(8):1168
22. Toqué F, Côme E, El Mahrsi MK, Oukhellou L (2016) Forecasting dynamic public transport Origin-Destination matrices with long-short term memory recurrent neural networks. In: 2016 IEEE

19th International conference on intelligent transportation systems (ITSC), pp 1071–1076

23. Zhu J, Feng F, Shen B (2018) People counting and pedestrian flow statistics based on convolutional neural network and recurrent neural network. In: 2018 33rd Youth academic annual conference of chinese association of automation (YAC), pp 993–998

24. Cho K, Van Merriënboer B, Bahdanau D, Bengio Y (2014) On the properties of neural machine translation: Encoder-decoder approaches. arXiv:14091259

25. Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv:14123555

26. Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. arXiv:14126980