Lifelong Learning Based Material Delivery Time Prediction for Helicopter Assembly

MA Lijun[1], *YANG Xianggui*[2], *GUO Yu*[1], *TONG Zhouqiang*[2], *HUANG Shaohua*[1*], *LIU Daoyuan*[1]

1. College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P. R. China;

2. China Aviation Industry Jiangxi Changhe Aviation Industry Co. Ltd., Jingdezhen 333000, P. R. China

(Received 5 September 2023; revised 21 December 2023; accepted 10 April 2024)

Abstract: The lack of key materials has emerged as one of crucial factors affecting the execution of helicopter assembly production plans. Accurate material delivery time prediction can guide assembly production planning and reduce frequent changes caused by material shortages. A lifelong learning-based model for predicting delivery time of materials is proposed on the basis of internal data sharing within the helicopter factory. During real-time prediction, the model can store new memories quickly and not forget old ones, which is constructed by gated recurrent unit (GRU) network layer, ReLU activation layer, and fully connected layers. To prevent significant precision degradation in real-time prediction, a regularization parameter constraint method is proposed to adjust model parameters. By using this method, the root mean square error (RMSE) in the model's prediction on the target domain data is reduced from 0.032 9 to 0.013 4. The accuracy and applicability of the model for real-time prediction in helicopter assembly is validated by comparing it with methods such as L2 regularization and EWC regularization, using 25 material orders.

Key words: helicopter assembly; material delivery forecast; lifelong learning; parameter regularization

CLC number: TP391 **Document code:** A **Article ID:** 1005-1120(2024)02-0147-11

0 Introduction

Due to various disturbance events, frequent adjustment of production plans due helicopter assembly line has become a prominent problem[1]. As one of the important influencing factors, the uncertainty in material delivery time will cause severe delays in material kitting time. Deviation between actual and expected time for material kitting time may result in reduced assembly efficiency[2]. The absence of critical materials may disrupt the existing production plan, potentially leading to downtime on the production line[3]. The "headquarters-branch" mode is currently the main model adopted by helicopter manufacturing enterprises, which means most of the key components of helicopters are produced in the group's own workshops. State data and statistical data are shared among different workshops. When making a production plan for helicopter assembly, the production planning department can directly obtain production process data, which provides comprehensive support in predicting material delivery time. The accuracy of real-time prediction of material delivery time is important for the execution of production plans and the optimization of scheduling costs.

According to literatures, predictive methods for material delivery time include support vector regression, decision tree regression, case-based reasoning, neural networks, etc. Lu et al.[4] proposed a dynamic scheduling model for aircraft assembly based on material delivery lead time prediction, uti-

*Corresponding author, E-mail address: shaohuah@nuaa.edu.cn.

How to cite this article: MA Lijun, YANG Xianggui, GUO Yu, et al. Lifelong learning based material delivery time prediction for helicopter assembly[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2024, 41(2): 147-157.

http://dx.doi.org/10.16356/j.1005-1120.2024.02.002

lizing the XGBoost algorithm to forecast delivery time and dynamically adjusting production plans based on the prediction results. However, this algorithm is not well-suited for processing high-dimensional sparse data. Chen et al.[5] proposed an integrated model by combining K-means clustering, feature selection, and the decision tree method into a single evaluation model to assess the performance of suppliers. Louvros et al.[6] used machine learning and case-based reasoning for real-time onboard prediction of the survivability of ships, but if there is no similar data available to the current issue, it may result in the inability to deduce new solutions. Ren et al.[7] developed a material delivery node prediction approach based on information entropy and used dynamic error compensation to refine the prediction results. However, building a state transition matrix and precisely describing the transition probabilities of each workstation state are particularly challenging in complex assembly scenarios. Lyu et al.[8] employed principal components analysis (PCA) and a 1D convolutional neural network (CNN) to predict the remaining life of aircraft engines, and Peng et al.[9] proposed a multi-input deep learning model for predicting terminal area traffic flows during convective weather. Although the aforementioned methods are effective in forecasting, they did not take into account the sequentiality of the data.

Material delivery time prediction involves predicting future delivery time using time series forecasting techniques. This can be achieved using both statistical and machine learning-based methods. Statistical-based methods, such as autoregressive models[10] and autoregressive moving average models[11], excel at processing linear and small-sample data. However, they may face challenges when dealing with large and non-stationary datasets due to their high computational costs and reliance on the assumption of data stationarity and linearity. Machine learning methods can be divided into traditional machine learning and deep learning. Traditional machine learning methods rely on manually selected features for modeling and prediction, such as linear regression models[12], support vector[13], decision trees[14], random forests[15], etc. However, for some time-series prediction problems with large data sets and complex features, the performance of traditional machine learning methods may be limited by the capabilities of the model.

Deep learning methods not only perform well in solving nonlinear and non-stationary data, but also handle prediction problems with large data sets and complex features. Ref.[16] proposed the first recurrent neural network model for time series prediction. However, the problem of vanishing or exploding gradients occurs with longer sequences. Long short-term memory (LSTM)[17] model solves the gradient problem, but it comes with a high computational cost and is susceptible to overfitting. Cho et al.[18] introduced the gated recurrent unit (GRU) model, addressing gradient issues while simplifying and facilitating training. Wang et al.[19] proposed a lightweight multi-layer residual temporal convolutional network model to target the highly complex kinematic and temporal correlation of human motion. Deep neural networks exhibit high prediction accuracy, but most of these methods are trained on fixed datasets. When the distribution of data changes, the model may experience significant prediction bias.

Although previous research provides technical support for improving the accuracy of material delivery time, real-time prediction under fluctuation in data distribution is relatively underexplored. Technologies such as intelligent recognition, advanced control, and intelligent sensing provide technical support for data collection in the workshop. The real-time data in the machining process of the machining workshop has a high dimensionality and is nonlinear. This paper analyzes the relevant factors that influence material delivery time, establishes a regression prediction model, and compares the prediction performance of popular time-series models on the dataset. To address the issue of reduced assembly efficiency caused by inaccurate material delivery time, a material delivery time prediction model is proposed. In practice, the operating law of the workshop changes dynamically over time, thus the data distribution representing the state of the material changes accordingly, and we add a lifelong learning approach to the model. This model has the ability to learn and accumulate knowledge over time within its neural network, surpassing the limitations of traditional prediction models that can only provide accurate predictions on similarly distributed data.

1 Description of the Problem

In this article, a specific Chinese helicopter manufacturing plant is studied, and Fig.1 illustrates the diagram of material delivery within the factory, where AGV denotes automated guided vehicle. The materials required for helicopter assembly are manufactured in the mechanic processing workshop and then delivered to the warehouse. The warehouse uses a large underground conveyor to distribute the matched materials to the corresponding assembly stations based on the delivery time nodes.

Define the remaining material delivery time as from the current moment until the material is delivered to the corresponding assembly station. The material remaining delivery time (MRDT) is defined as

$$\mathrm{MRDT}=T_{\mathrm{m}}+T_{\mathrm{c}}+T_{\mathrm{s}} \tag{1}$$

where T_{m} represents the remaining time to completion, T_{c} the time required for delivery from mechanic processing workshop to warehouse, and T_{s} the time of transportation via conveyor belt. As transportation route and speed remain constant, we consider the time required for material delivery as a constant value.

The warehouse system only maintains records

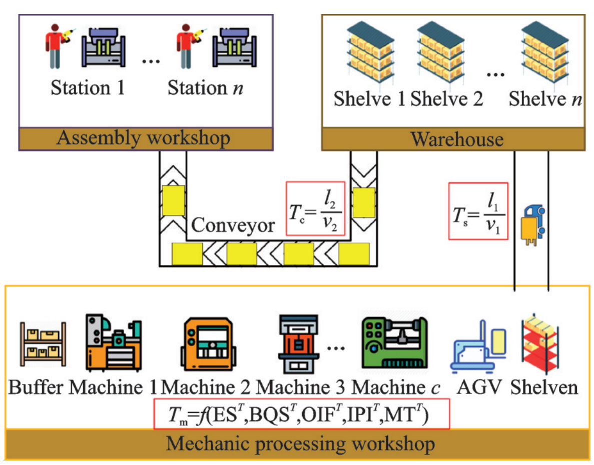

Fig.1 Diagram of material delivery in a helicopter manufacturing plant

of the current status of each inventory item and cannot accurately show the delivery time for non-inventoried materials. The precision and promptness of material delivery are vital to the efficient operation of assembly tasks. Predicting the delivery time of materials beforehand can transform management mode form post-adjustments to pre-adjustments.

The materials required are processed by the corresponding machining workshop, which has machining equipment with a total number of M. The primary objective of this article is to forecast critical shortages of essential materials which were frequently encountered during statistical analysis in history. The duration of storage in the warehouse after the completion of material processing is not taken into account for the remaining delivery time.

The machining process discussed in this article follows the principle of processing different materials along predetermined routes, with each machine processing only one material at a time. During equipment operation, incoming materials are required to be queued in the input buffer area waiting for processing. Upon completion of the machining process, the finished parts will directly enter the output buffer area, awaiting transportation to the next machine facility. The input and output buffer areas operate on a principle of first in first out for processing. There are five main factors that influence the delivery time of materials.

The uncertain equipment status (ES) of the machine can impact the required machining time. Considering the sufficient buffer capacity and ample number of AGVs, the waiting time due to material transport is eliminated. Hence, only the machine status is considered during data collection.

$$\mathrm{ES}^{T}=[\mathrm{MU}_{n}^{T},\mathrm{CW}_{n}^{T}] \tag{2}$$

where ES^{T} denotes equipment operating status at time T, n machine number, MU_{n}^{T} average machine utilization rate from the nearest completed material order to the current time T, and CW_{n}^{T} machine n's continuous working time at time T.

BQS represents the storage status of the queue in the buffer area, which affects the waiting time of materials. The queue information reflects details regarding the processing route and order of materials. The BQS queue information is shown as

$$\mathrm{BQS}^T=\begin{bmatrix}\mathrm{IC}_{n,1}^T,\mathrm{IC}_{n,2}^T,\cdots,\mathrm{IC}_{n,M}^T\\ \mathrm{OC}_{n,1}^T,\mathrm{OC}_{n,2}^T,\cdots,\mathrm{OC}_{n,M}^T\end{bmatrix} \tag{3}$$

where BQS^T represents the queue information for the buffer area of all machine tools at time T, $\mathrm{IC}_{n,i}^T$ the type of the ith material entering buffer area queue for machine tool n at time T, and $\mathrm{OC}_{n,i}^T$ the type of the ith material leaving the buffer area queue for machine tool n at time T.

The order information (OIF) describes the composition of materials involved in the processing order, including material identification and quantities. The number and type of materials in the order play a determining role in the overall processing time of the order.

$$\mathrm{OIF}=[M_1,M_2,\cdots,M_X] \tag{4}$$

where M_i represents the type of the ith material and X the number of material type in the workshop.

In-process information (IPI) is determined by the type of work-in-progress and the accumulated processing time for the work-in-progress on the machine, which in turn determines the remaining processing time on that machine. TM_n^T represents the type of material being processed on machine n at time T, while PT_n^T represents the duration of processing that material on machine n up to time T.

$$\mathrm{IPI}^T=[\mathrm{TM}_n^T,\mathrm{PT}_n^T] \tag{5}$$

In addition, statistics on the completion of the current order are also required, including the type of material quantities already completed and the remaining processes of unfinished processing, which is shown as

$$\mathrm{MT}^T=[N_c^T,P_c^T] \tag{6}$$

where MT^T represents the order completion status at time T, N_c^T the process of material c being completed at time T and P_c^T the remaining processing steps of material c at time T.

Therefore, the feature dataset required for material delivery time prediction model can be represented by

$$\mathrm{FD}=[\mathrm{ES},\mathrm{BQS},\mathrm{OIF},\mathrm{IPI},\mathrm{MT}] \tag{7}$$

where FD represents the feature set of predicted material delivery time. And a deep neural network is utilized to perform regression prediction on material delivery time (MDT) using the features extracted from FD.

$$\mathrm{MDT}=f(\mathrm{FD}) \tag{8}$$

Eq.(7) represents the mapping relationship between material delivery time and the data features.

2 Material Delivery Time Prediction Model

In order to solve the problems of uncertainty of material delivery time in the shop floor and the easy failure of fixed models in real-time prediction, we propose a lifelong learning-based framework. The architecture of the material delivery time prediction model is based on the GRU network and incorporates lifelong learning to address the issue of reduced prediction accuracy caused by data distribution fluctuations over time. The model's structure is illustrated in Fig.2. Three major steps are involved in the lifelong learning-based model for material delivery time prediction. The first step, the data that characterizes the state of the workshop is collected. The prediction model is trained by using historical data of the workshop through the gradient descent algorithm. By repeatedly conducting training rounds, the most accurate predictive model parameters can be attained. The predictive model utilizes real-time data for application validation. We monitor the root mean square error (RMSE) during the real-time prediction process and use it to determine whether the model's error exceeds a threshold. The third step is to fine-tune the parameters of the model when the error exceeds the baseline. The source model is used to initialize the target mode and compute the importance of each parameter. The regularization constraint is used to retrain the model, and the trained model is updated after the training is completed.

In that case, define the data before the distribution changes as the source domain, and after the change as the target domain. The relevant information from the source domain is preserved to constrain the parameter changes in the training of the target domain, thus enabling the model to make predictions on data with different distributions.

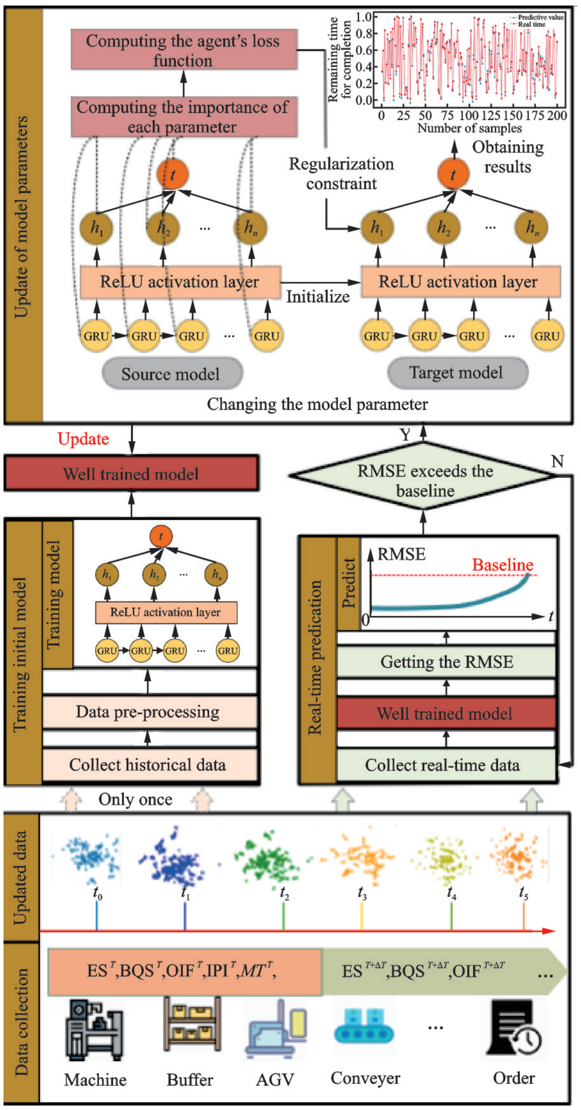

Fig.2 Material delivery time prediction framework

Feature scaling is utilized as a crucial data preprocessing step to normalize features with disparate scales, which can help to alleviate the imbalanced sample distributions. The defining formula is shown as

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{9}$$

where x' is the normalized value of the data and x the data before normalization; $x_{\max}$ and $x_{\min}$ denote the maximum and minimum values of the corresponding features, respectively.

A time series prediction model is specifically designed for the delivery forecasting of materials, as it involves the regression prediction of time series data. GRU is not only able to learn quickly, but also suitable for scenarios especially when the sequences are relatively short. GRU is a type of recurrent neural network that focuses on addressing the issues of vanishing and exploding gradients, and can also solve the problem of information loss in traditional RNN networks. Its main characteristic is the introduction of gate mechanisms. Through gate mechanisms, GRU can selectively "retain" or "forget" input data, thus achieving remembering and forgetting of information. In GRU, the state of each unit can be weighted by the controller, which includes update gate ($\boldsymbol{z}_t$), reset gate ($\boldsymbol{r}_t$), and candidate state ($\tilde{\boldsymbol{h}}_t$).

The update gate functions control the extent to which input data modifies the current state, with a higher value indicating a stronger incorporation of previous state information.

$$\boldsymbol{z}_t = \sigma(\boldsymbol{W}_z \boldsymbol{x}_t + \boldsymbol{U}_z \boldsymbol{h}_{t-1} + \boldsymbol{b}_z) \tag{10}$$

The reset gate is used to control the impact of the unit's historical information on the current state. The smaller the reset gate value is, the less previous state information is being incorporated. If the correlation between the previous state information and the current input is weak, the reset gate is triggered.

$$\boldsymbol{r}_t = \sigma(\boldsymbol{W}_r \boldsymbol{x}_t + \boldsymbol{U}_r \boldsymbol{h}_{t-1} + \boldsymbol{b}_r) \tag{11}$$

Once the update gate and the reset gate are computed, the GRU calculates the candidate state by using

$$\tilde{\boldsymbol{h}}_t = \tanh\left(\boldsymbol{W}_t \boldsymbol{x}_t + \boldsymbol{U}_t\left(\boldsymbol{r}_t \odot \boldsymbol{h}_{t-1}\right) + \boldsymbol{b}_t\right) \tag{12}$$

The output of GRU at time step t is given as

$$\boldsymbol{h}_t = (\boldsymbol{I} - \boldsymbol{z}_t) \odot \boldsymbol{h}_{t-1} + \boldsymbol{z}_t \odot \tilde{\boldsymbol{h}} \tag{13}$$

where x_t represents the input vector, h_{t-1} the hidden state vector from the previous time step, z_t the update gate vector, r_t the reset gate vector, $\tilde{h}_t$ the new candidate hidden state vector, h_t the hidden state vector at the current time step, and $\sigma(\cdot)$ the sigmoid function, whose range falls within $(0, 1)$. $\odot$ denotes an element-wise multiplication operation. W, U and b represent the weight matrices for input, transition from the previous time step's hidden state vector, and the bias vector, respectively.

GRU has shown great predictive performance in sequential data prediction. It is usually trained on a dataset and then its network parameters are frozen before it is deployed in the target application. As time elapses and the distribution of data changes, the performance of neural networks may deteriorate. To adapt to changes in data distribution, it is necessary to make adjustments to the parameters of the neural network for mitigating the risks of overfitting and catastrophic forgetting. If source domain data and target domain data are mixed during model adjustments, a large amount of storage space will be consumed.

Artificial intelligence predominantly relies on fixed datasets and stationary environments. In general, the input data to the model consists of production data obtained from static conditions[20]. However, these data need to be carefully shuffled, balanced, and homogenized before presented to the model. When GRU is used for real-time prediction, in some cases, the model may underperform or experience rapid performance degradation on previously learned tasks. Lifelong learning could enable models to adapt to the distribution of real-time data changes. Regularization-based methods can effectively avoid the issue of large storage space consumption. When the model experiences a decrease in prediction accuracy, abstract historical information can be saved and used for retraining the network to avoid catastrophic forgetting. The principle is to evaluate the importance of neural network parameters trained on source domain, and prevent significant changes in important parameters during training on the target domain. Gu et al.[21] visualized the distribution of parameter importance through experiments, and proposed a method based on Taylor expansion to assess the significance of neurons. They concluded that the importance of a neuron is calculated by multiplying the absolute value of its activation value with the gradient of the loss function concerning its activation value. By applying the aforementioned idea, the importance of the parameter can be assessed by analyzing its impact on the overall performance of the model. The importance of each parameter is measured by the absolute value of the product of the parameter's magnitude and its gradient with respect to the loss function after completing training in the source domain. This yields the parameter importance matrix shown as

$$W(H)=\left|\boldsymbol{\theta}\odot\frac{\delta L(H)}{\delta\boldsymbol{\theta}}\right| \tag{14}$$

where H denotes all parameters in the network, $\boldsymbol{\theta}$ the parameter matrix, and $\frac{\delta L(H)}{\delta\boldsymbol{\theta}}$ the gradient matrix of the loss function $L(\cdot)$ with respect to $\boldsymbol{\theta}$. After calculating the importance of parameters, we can use them to impose constraints on different parameters in training of the target domain. Additionally, the original loss function is replaced with a proxy loss function in the target domain shown as

$$L_2=L_2'+C*\sum_K W(H)*(\tilde{\theta}_k-\theta_k)^2 \tag{15}$$

$$C=\frac{L_2'}{\sum_k W(H)*(\tilde{\theta}_k-\theta_k)^2+\xi} \tag{16}$$

where L_2 represents the proxy loss function, L_2' the original loss function on the target domain, and C the penalty term and dynamically adjustment of the value to balance the penalty and original loss functions. ξ is an additional damping factor that prevents the occurrence of excessively large or small values of C. $\tilde{\theta}_k$ denotes the parameter values after training on source domain, while θ_k denotes those used for training on target domain. By utilizing the proxy loss function and target domain data, we retrain the model by initializing the parameters with the trained parameters of the source domain and update them using the gradient optimization. The model generated from this approach possesses the capability to cater to the prediction requirements of both the source and target domains, constantly accumulating knowledge via this methodology, and eventually extending predictions to encompass diverse data distributions within the workshop. The parameter update process of the model is illustrated in Fig.3.

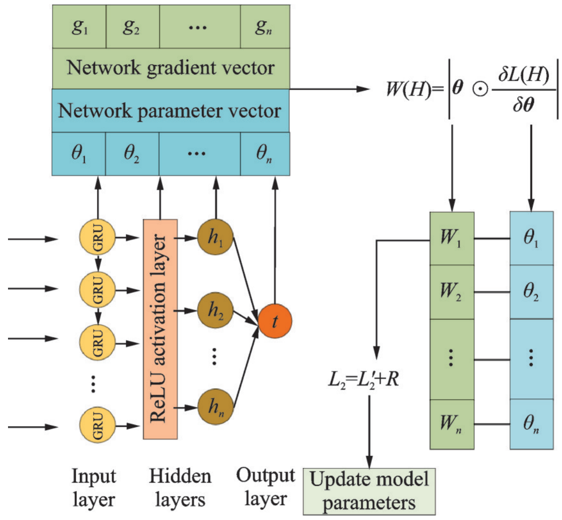

Fig.3 Model parameter update process

3 Experimental Application and Analysis

As an instance, information relevant to the materials necessary for the assembly of a specific helicopter is gathered. 25 material orders corresponding to the frame, beam, and horizontal components located at the mid-end of the fuselage are selected for validation. Utilizing the data pertaining to materials required, a set of 15 material orders are selected as the source domain, while 10 orders are chosen as the target domain.

The dataset is preprocessed, and through comparative experiments, the GRU neural network model shows the best performance in predicting the delivery time of materials. The optimal solution for hyperparameters is through the approach of a tree-structured Parzen estimator (TPE). It is a Bayesian optimization algorithm based on tree structure, which is used to solve the global optimization problem of black box function. As follows, epoch=100, batch size=256, learning rate=0.000 1, time step = 6, and the threshold ξ = 0.000 002 84. The hidden layer structure of the neural network is 400-215. The Adam optimizer is employed to adjust the parameters in the model. During training, the loss function is measured using RMSE, which is in line with the original data unit and makes it easier to interpret the performance of the model on the data. The formula for RMSE is shown as

$$\mathrm{RMSE}=\sqrt{\frac{1}{N}\sum_{i=1}^{n}(Y_i-f(x_i))^2} \tag{17}$$

where the notation Y_i represents the true values, $f(x_i)$ the predicted values, and N the total number of data in the batch.

To comprehensively evaluate the performance of the model, we also use R-square (R^2), mean absolute error (MAE), and symmetric mean absolute percentage error (SMAPE) as evaluation metrics. R^2, also referred to as the coefficient of determination, reflects the degree to which the independent variable explains the variation in the dependent variable. The closer the value approaches 1, the higher the accuracy of the model's fit. The MAE represents the anticipated value of absolute error loss, whereas SMAPE serves as another performance metric. Smaller values for these measures indicate superior model performance. The formulas for calculating R^2, MAE and SMAPE are given as

$$R^2=1-\frac{\sum_{i=1}^{n}(Y_i-f(x_i))}{\sum_{i=1}^{n}(Y_i-\overline{Y})} \tag{18}$$

$$\mathrm{MAE}=\frac{1}{n}\sum_{i=1}^{n}\left|(Y_i-f(x_i))\right| \tag{19}$$

$$\mathrm{SMAPE}=\frac{1}{n}\sum_{i=1}^{n}\frac{|f(x_i)-Y_i|}{(|f(x_i)|+|Y_i|)/2}\times 100\% \tag{20}$$

The advantage of GRU predictions is validated by comparing its performance to other time series forecasting models available on the dataset. The predicted results after training are presented below. Fig.4 illustrates the loss function values for various models on the validation set during the training process. The x-axis represents the number of training iterations, while the y-axis the value of RMSE. The results are obtained by training four different models, namely transformer, temporal convolutional network (TCN), deep neural network (DNN), and GRU. From Fig.4, it can be observed that the GRU exhibits faster convergence rate on the given dataset. Furthermore, the convergence process is relatively stable. The predictive performance of TCN is comparable to that of DNN. However, the transformer exhibits the poorest predictive efficiency. Consequently, GRU has a significant advantage in predicting material delivery time.

Table 1 shows the metrics of each model on the source and target domain validation sets. It can be observed that RMSE of GRU model is 0.006 1 on the source domain validation set and 0.032 9 on the target domain validation set, which are smaller than those of other models. Most evaluation metrics of the GRU model are superior to those of other models. This indicates that the GRU model has high prediction accuracy and good generalization performance on the given dataset. The predictive performance of all models on the target domain is relatively poor. The RMSE of GRU in the validation set increases from 0.006 1 to 0.032 9, indicating a growth rate of 439% and suggesting that utilizing a fixed-parameter model for real-time material delivery time prediction is not a viable option. The data that represents the state of the workshop undergoes dynamic changes in distribution over time.

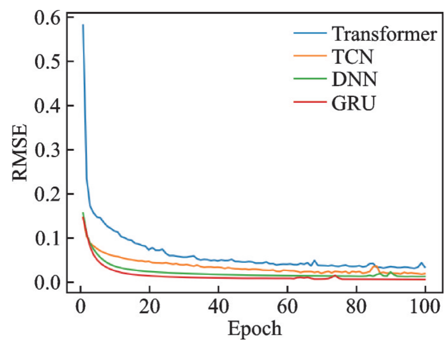

Fig.4 Comparison of predictive performance across different methods

Table 1 Prediction of different methods on source and target domain data sets

Model	Source domain validation data				Target domain validation data			
	RMSE	R^2	MAE	SMAPE/%	RMSE	R^2	MAE	SMAPE/%
Transformer	0.032 2	0.987 3	0.025 4	12.98	0.070 8	0.940 6	0.055 7	24.90
DNN	0.012 2	0.998 5	0.007 7	4.02	0.033 5	0.986 7	0.024 3	10.82
TCN	0.017 6	0.996 2	0.012 9	8.51	0.084 1	0.912 1	0.072 8	17.50
GRU	0.006 1	0.999 6	0.005 0	3.22	0.032 9	0.987 0	0.023 3	10.27

The application of the uniform manifold approximation and projection (UMAP) algorithm to two distinct datasets results in the reduction of their dimensionality, allowing for comparison of their distributions. This analysis reveals significant differences, which are illustrated in Fig.5. This highlights the importance of adjusting the real-time delivery prediction model in response to the magnitude of its bias. In this setting, lifelong learning comes as a natural solution.

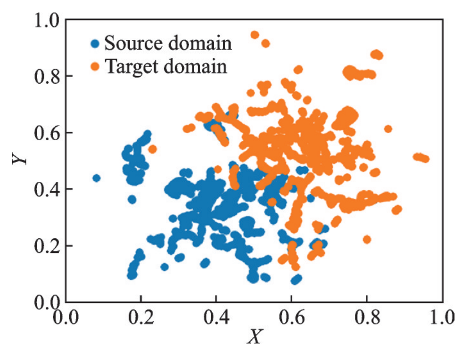

Fig.5 Data distribution of source and target domains

The concept of lifelong learning is incorporated into parameter updates to prevent catastrophic forgetting of previous data distributions. Preserving the parameters and gradients of the original model and computing the importance matrix of the parameters, the model is then retrained on the target domain via a proxy loss function. In Table 2, GRU_{PGC} refers to using parameter P and gradient G to calculate the importance matrix, and using the adaptive variable C as a penalty coefficient. GRU_{PG} refers to the result of using a constant instead of the adaptive variable. GRU_{L2} refers to using L2 regularization to constrain the model parameters. GRU_{EWC} refers to using the elastic weight consolidation (EWC) to constrain the model parameters. GRU_{SI} refers to using synaptic intelligence (SI)[22] to constrain the model parameters. GRU_{MAS} refers to using memory aware synapses (MAS)[23] to constrain the model parameters.

Based on the results shown in Table 2, it can be observed that after adjustment, the GRU_{PGC} has a validation set RMSE of 0.013 3 on the source domain and a validation set RMSE of 0.012 4 on the target domain. The model not only adapts to the data distribution of the target domain but also does not forget the source domain. The L2 algorithm has the worst solution performance, followed by EWC, SI, MAS and then PG, indicating that the PGC method can better maintain the stability of the model's prediction accuracy on this dataset.

Table 2 Prediction of different regularization methods on source and target domain datasets

Metric value	Target domain data				Source domain data			
	RMSE	R^2	MAE	SMAPE/%	RMSE	R^2	MAE	SMAPE/%
GRU_{PGC}	0.013 4	0.997 8	0.008 4	5.02	0.013 3	0.997 9	0.010 8	6.34
GRU_{PG}	0.013 8	0.997 7	0.008 8	5.00	0.013 4	0.997 9	0.010 8	8.92
GRU_{L2}	0.015 3	0.997 2	0.009 8	5.43	0.014 3	0.997 6	0.011 4	7.22
GRU_{EWC}	0.013 2	0.997 9	0.008 1	4.64	0.014 5	0.997 5	0.011 5	6.42
GRU_{SI}	0.014 1	0.997 6	0.009 2	5.37	0.015 2	0.997 2	0.012 2	8.01
GRU_{MAS}	0.014 2	0.997 6	0.009 0	5.32	0.013 4	0.997 9	0.010 8	6.74

Fig.6 shows the descent of the loss function of the model on the target domain training set (represented by LOSS 1), validation set (represented by VLOSS 11). The descent of the loss function on the source domain is represented VLOSS 10. During training with regularized constraints, a gradual ascent of the loss function on the source domain is followed by a decrease in the loss function on the target domain, until they both converge to similar values. Regularization constraints result in a balanced solution space for model parameters, ensuring accurate predictions for both source and target domains.

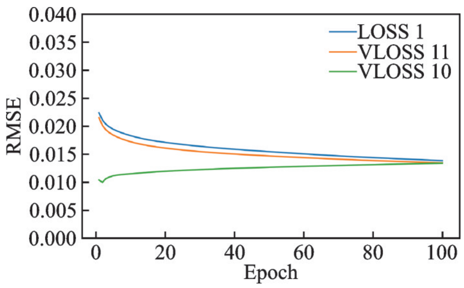

Fig.6 GRU_{PGC} loss functions for training and validation sets on source and target domains

Figs.7—9 depict the number of sample points on the horizontal axis and the corresponding delivery time on the vertical axis, with triangular markers indicating predicted values and red markers indicating actual values. Validation 1 represents the model's prediction performance on the validation set of the target domain without training with regularized constraints. Fig.7 shows that the results of this model have many deviations. Validation 11 and Validation 10 represent the prediction performances after completing training with regularized constraints on the target domains. It can be seen that regularized constraints are highly effective in avoiding catastrophic forgetting.

The following conclusions can be drawn from the aforementioned experiments:

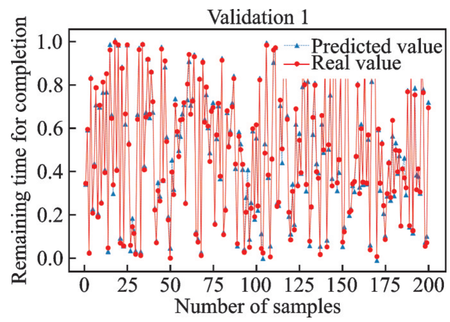

Fig.7 Validation effect of GUR_{PGC} on the target domain after training on the source domain

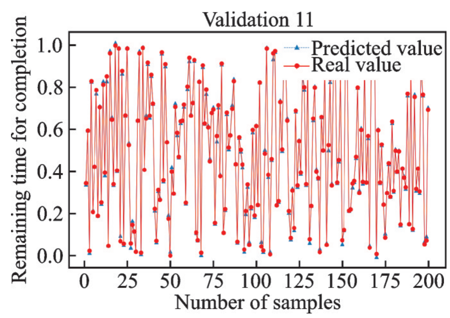

Fig.8 Validation effect of GUR_{PGC} on the target domain after training on the target domain

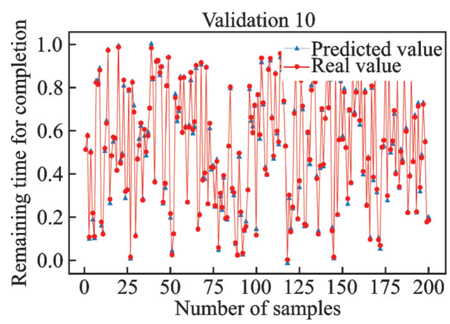

Fig.9 Validation effect of GUR_{PGC} on the source domain after training on the target domain

(1) In time series prediction models, GRU yields higher accuracy in predicting material delivery time in the studied helicopter workshop.

(2) In practice, the state data of the workshop dynamic changes over time. Fixed parameter models experience a gradual decrease in accuracy in real-time prediction as the data distribution changes.

(3) The proposed method has the advantage of utilizing adaptive parameter C to trade-off between the allowed forgetting and the new task loss. During the training, the model can learn the importance weights. When learning the new distribution data, changes to important parameters are penalized.

(4) Incorporating the concept of lifelong learning, the prediction model exhibits high reliability and adaptability. This provides data support to achieve coordination among material processing, component assembly, and production planning departments, thereby enhancing the feasibility and achievability of helicopter assembly plans.

4 Conclusions

This paper focuses on predicting delivery time for helicopter assembly materials with a GRU-based model incorporating lifelong learning. First, by comparing the prediction effects of different time-series prediction models, it is concluded that GUR predicts well on material prediction. Then, it is confirmed that the distribution of data changes dynamically with time by using dimensionality reduction. Finally, we compare different regularized lifetime learning methods. By utilizing regularized constraints to fine-tune model parameters, the model can not only make predictions on new data, but also prevent catastrophic forgetting. The predicted results can be used to guide the production plan in the assembly workshop, reduce the frequency of changes to the production plan to some extent, lead to a pre-intervention mode and improve production efficiency in the workshop. In future work, the model's network structure can be pruned or expanded based on predicted data to ensure that the model completes predictions with the smallest possible structure.

References

[1] BIN L, QIANG L. Dynamic scheduling of aircraft mobile production line considering material supply interference[J]. Journal of Beijing University of Aeronautics and Astronautics, 2020, 46(8): 1521-1534.

[2] HANSON R, BROLIN A. A comparison of kitting and continuous supply in in-plant materials supply[J]. International Journal of Production Research, 2013, 51(4): 979-992.

[3] CAO Yuanchong, XIONG Hui, ZHUANG Cunbo, et al. Dynamic scheduling of complex product discrete assembly workshop based on digital twin[J]. Computer Integrated Manufacturing Systems, 2021, 27(2): 557-568. (in Chinese)

[4] LU Bin, LU Zhiqiang, ZHANG Yongfeng. Dynamic scheduling problem of aircraft assembly based on material delivery date prediction[J]. Computer Integrated Manufacturing Systems, 2022, 28(9): 2939-2952. (in Chinese)

[5] CHEN Y S, CHENG C H, LAI C J. Extracting performance rules of suppliers in the manufacturing industry: An empirical study[J]. Journal of Intelligent Manufacturing, 2012, 23(5): 2037-2045.

[6] LOUVROS P, STEFANIDIS F, BOULOUGOURIS E, et al. Machine learning and case-based reasoning for real-time onboard prediction of the survivability of ships[J]. Journal of Marine Science and Engineering, 2023, 11(5): 890.

[7] REN Yinghui, HUANG Xiangming, MA Zhongkai, et al. Time node prediction method for material delivery based on information entropy[J]. China Mechanical Engineering, 2018, 29(22): 2725-2732. (in Chinese)

[8] LYU D F, HU Y W. Remaining life prediction of aeroengine based on principal component analysis and one-dimensional convolutional neural network[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2021, 38(5): 867-875.

[9] PENG Y, WANG H, MAO L M, et al. Flow prediction method for terminal area in convective weather based on deep learning[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2021, 38(4): 634-645.

[10] ZHANG Rui, JIA Hu. Production performance forecasting method based on multivariate time series and vector autoregressive machine learning model for waterflooding reservoirs[J]. Petroleum Exploration and Development, 2021, 48(1): 175-184. (in Chinese)

[11] BENVENUTO D, GIOVANETTI M, VASSALLO L, et al. Application of the ARIMA model on the COVID-2019 epidemic dataset[J]. Data in Brief, 2020, 29: 105340.

[12] ZHANG F J, WANG S L, LIANG J, et al. Study on multiple linear regression prediction of bending fatigue life of steel wire rope[C]//Proceedings of Journal of Physics: Conference Series. [S.l.]: IOP, 2022, 2230(1): 012007.

[13] WANG Lei, ZHANG Ruiqing, SHENG Wei, et al. Regression prediction and anomaly data detection based on support vector machine[J]. Chinese Journal of Electrical Engineering, 2009, 29(8): 92-96. (in Chinese)

[14] LIU C, HU Z, LI Y, et al. Forecasting copper prices by decision tree learning[J]. Resources Policy, 2017, 52: 427-434.

[15] WEI Qin, CHEN Shijun, HUANG Weibin, et al. Spot market clearing price prediction method using random forest regression[J]. Proceedings of the CSEE, 2021, 41(4): 1360-1367,1542. (in Chinese)

[16] FANG W, CHEN Y, XUE Q. Survey on research of RNN-based spatio-temporal sequence prediction algorithms[J]. Journal on Big Data, 2021, 3(3): 97-110.

[17] YU Y, SI X, HU C, et al. A review of recurrent neural networks: LSTM cells and network architectures[J]. Neural Computation, 2019, 31(7): 1235-1270.

[18] CHO K, VAN MERRIENBOER B, GULCEHRE C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[EB/OL]. (2014-09-03). https://doi. org/10.48550/arXiv.1406.1078.

[19] WANG Y,QIAO B. A lightweight temporal convolutional network for human motion prediction[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2022, 39(S1): 150-157.

[20] GANG Y, XIAO L, CHANG Z, et al. Multi-objective coupling optimization of electrical cable intelligent production line driven by digital twin[J]. Robotics and Computer-Integrated Manufacturing, 2024(86): 102682.

[21] GU S, FENG Y. Investigating catastrophic forgetting during continual training for neural machine translation[EB/OL]. (2020-11-30). https://doi.org/10.48550/arXiv.2011.00678.

[22] ZENKE F, POOLE B, GANGULI S. Continual learning through synaptic intelligence[C]//Proceedings of International conference on machine learning. [S.l.]: PMLR, 2017: 3987-3995.

[23] ALJUNDI R, BABILONI F, ELHOSEINY M, et al. Memory aware synapses: Learning what (not) to forget[C]//Proceedings of the European Conference on Computer Vision (ECCV). [S.l.]:[s.n.], 2018: 139-154.

Authors Ms. **MA Lijun** received the B.S. degree in Mechanical Engineering from Shaanxi University of Science and Technology. She is currently a graduate candidate at College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics (NUAA). Her research interests focus on industrial big data and digital manufacturing.

Dr. **HUANG Shaohua** received the B.S. and M.S. degrees in Mechanical Manufacturing and Automation from Xinjiang University in 2011 and 2014, respectively, and the Ph.D. degree in Aerospace Manufacturing Engineering at NUAA in 2019. From 2020 to 2022, he worked as a postdoctoral fellow in the Department of Industrial Engineering at Tsinghua University. He joined in NUAA in July 2022, where he is an assistant professor of Intelligent Manufacturing Laboratory. His current research interests include the Internet of things, industrial big data, and production process optimisation.

Author contributions Ms. **MA Lijun** built the model, conducted the analysis, interpreted the results and wrote the manuscript. Mr. **YANG Xianggui** provided training data for the recurrent network model. Prof. **GUO Yu** designed and conceptualized the study. Mr. **TONG Zhouqiang** provided data for the analysis of lifelong learning. Dr. **HUANG Shaohuang** conducted the experimental design. Dr. **LIU Daoyuan** participated in the discussion of the research background. All authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing interests.

(Production Editor: WANG Jing)

基于终身学习的直升机装配车间物料送达时间预测

马立俊[1],阳祥贵[2],郭 宇[1],童周强[2],黄少华[1],刘道元[1]

(1.南京航空航天大学机电学院,南京 210016,中国; 2.江西昌河航空工业有限公司,景德镇 333000,中国)

摘要:关键物料的缺失已成为影响直升机装配生产计划执行的关键因素之一。准确的物料送达时间可指导装配生产计划的制定,一定程度上避免了由缺料导致的生产计划频繁变更。在直升机车间内部数据共享的基础上,一种基于终身学习的物料送达时间预测模型被提出。该模型由门控循环单元(Gated recurrent unit, GRU)网络层、ReLU激活层和全连接层构成,在实时预测时,可快速存储新的记忆且不遗忘旧的。为避免在实时预测中的精度大幅度降低,一种正则化的参数约束方式被提出来对模型参数进行调整。该方法的应用使得模型在目标域数据上的预测误差从0.032 9降低到0.013 4。使用25个物料清单数据进行模型验证。通过与L2正则化、EWC正则化等常用的正则化方法进行对比,验证了所建立的模型在实时预测上的准确性与实用性。

关键词:直升机装配车间;物料送达预测;终身学习;参数正则化