DOI 10.15276/imms.v13.no3-4.195
UDC 004.93'11:004.942

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 3-4, pp. 195-203

USE OF PRE-TRAINED NEURAL NETWORKS FOR MODELING NONLINEAR DYNAMIC OBJECTS

A.A. Orlov

National Odesa Polytechnic University
1 Shevchenko Ave., Odesa, 65044, Ukraine
email: andrey.orlov.od@gmail.com

The paper considers a class of problems of identification of nonlinear dynamic objects with continuous characteristics using neural networks with time delays. The multiple use of pre-trained neural networks to identify objects of different nature with similar laws of functioning is substantiated. The aim of this work is to reduce the training time of neural network models without significant loss of accuracy by developing a method for pre-training neural networks with time delays in the tasks of identifying nonlinear dynamic objects with continuous characteristics. The scientific novelty of the work is the further development of the method of pre-training neural networks with time delays in the tasks of identifying nonlinear dynamic objects with continuous characteristics, which allows reducing the training time of neural network models without significant loss of accuracy. The method consists in extracting general patterns from the base dataset at the pre-training stage and using them to solve specific problems at the stage of retraining models on the target dataset. A formal criterion is proposed to determine the moment of termination of the neural network pre-training, the use of which allows avoiding retraining of the base model and ensuring a significant reduction in the model training time on the target dataset. The practical significance of the work is to develop an algorithm for the method of pre-training neural networks with time delays in the tasks of identifying nonlinear dynamic objects with continuous characteristics, which reduces the training time of neural network models and the loss of model accuracy. To study the convergence rate of the training algorithm and modeling accuracy, an experiment was conducted with test nonlinear dynamic objects. The obtained modeling results demonstrate the effectiveness of the proposed method. The value of this study is *to* determine the area of effective use of the proposed method, namely, when the general and target datasets do not have significant differences and the target dataset is of sufficient size to reflect the properties of the research object.

Keywords: nonlinear dynamic objects, modeling, neural networks with time delays, pre-training.

Introduction. In the modern world, against the background of rapid technological development, the issues of effective modeling of modern management objects are becoming increasingly relevant. Classical methods, although reliable and widely used, are increasingly proving to be insufficiently effective for representing complex objects of the world [1, 2]. Complex control objects are characterized by a high degree of nonlinearity and dynamics, the ability to adapt to various environmental conditions and operating requirements.

Due to the above requirements, researchers and engineers prefer to consider complex objects as a "black box" whose internal structure and functioning algorithms are not accessible to an external observer [1, 3]. Such objects are well described by simulation models that approximate or mimic the behavior of a real system, which is difficult or impossible to express analytically. Simulation models are built on the basis of an input/output experiment, in which an object is analyzed based on its response (output signals) to external influences (input signals).

Today, neural networks are widely used as simulation models [2]. In the world of rapidly evolving technologies, neural networks have become an essential tool in the field of artificial intelligence, demonstrating impressive capabilities in solving complex tasks ranging from pattern recognition and natural language synthesis to identifying simulation models.

Such models can be trained to simulate complex behavior based on the observation of input and output data, which is why they are often used to model time dependencies and predict the behavior of complex dynamic systems.

Despite advances in the field of neural networks, the problems of effective learning with limited data and computational limitations remain unresolved [4]. Therefore, the task of developing methods to improve the characteristics of neural networks remains relevant.

Analysis of research and publications. Different approaches are used to overcome the limitations that hinder the development of neural networks in the field of complex object identification, depending on the application, object properties, availability and amount of training data. Having systematized the research in the field of improving neural network training methods, the following key areas can be identified.

Improvement of learning methods aimed at creating models capable of accelerating learning and quickly adapting to new tasks based on limited data. This area includes experiments with activation functions [5], learning methods [4], and the structure of neural networks [6].

The development of learning methods includes adaptive optimization algorithms such as Adafactor, LAMB, Ranger [7], which can more effectively use gradients for learning, as well as techniques for thinning layers and parameters in neural networks, which help reduce the number of calculations and speed up the learning process. This approach is implemented using the Sparse Training and Magnitude Pruning methods [8].

The choice of network architecture is carried out using automatic machine learning (AutoML) methods, which significantly speeds up the model development process. Such methods include Neural Architecture Search (NAS) [6, 9].

The construction of surrogate models makes it possible to reduce the size of neural network models and speed up computations without significant loss of accuracy, which is especially relevant given the growing demands on computing resources and the use of neural networks on mobile devices. This area includes such algorithms as Deep Compression [10], the construction of linear [3] and integral [11] surrogate models.

Pre-training allows neural networks to extract information from large amounts of data before the main training phase and can accelerate model convergence during training on target data, increasing their performance even on datasets of limited size [12].

A variation of pre-training is the transfer learning technique, which allows using the knowledge gained from solving one problem to improve the solution of another [13]. This approach is actively used when the available training data for a new task is limited, and it can significantly speed up learning and improve the generalizability of the model.

Modeling practice is well known for cases when you have to solve similar tasks that are repeated in different fields of activity due to the fact that objects of different nature have similar laws of functioning. In the field of software development, such tasks have long been successfully solved by reusing once-written code in the form of function libraries, classes, or a framework, depending on the degree of generalization of the task.

In the emerging field of simulation modeling, this approach is just beginning to be used. In this case, a pre-trained neural network can be used as a reusable solution. In this case, the construction of the target model consists of extracting general patterns from the base dataset at the pre-training stage and using them to solve specific problems at the stage of training models on the target dataset [12].

The described technique has several advantages. First, it allows using large amounts of data to extract general patterns, which is especially useful in the absence of sufficient data in the target dataset. Secondly, pre-trained models can be successfully applied to different tasks, even if they are related to different areas. This saves time and resources, as there is no need to train the model from scratch for each new task. Thus, thanks to pre-training, neural networks are able to demonstrate impressive results even in conditions of limited resources.

This approach is a fairly common and effective practice for reducing the training time of convolutional neural networks and building universal models before they are retrained on a specific task. VGG (Visual Geometry Group) pre-trained convolutional networks are widely known and have been successfully used for image classification tasks, GPTs are designed for text generation.

At the same time, there is a lack of work in the field of pre-training neural networks that model the behavior of nonlinear dynamic objects with continuous characteristics.

This article discusses methods of improving the characteristics of neural networks based on the idea of pre-training as a promising direction of identification of complex continuous objects, which is dynamically developing and able to effectively cope with the requirements of modern modeling tasks.

The aim of this work is to reduce the training time of neural network models without significant loss of accuracy by developing a method for pre-training neural networks with time delays in the tasks of identifying nonlinear dynamic objects with continuous characteristics.

Formulation of the research problem. The formal statement of the problem of pre-training neural networks can be formulated as follows.

Let S be an initial problem for which there is a large amount of labeled data (dataset D_S):

$$D_S = \{(x_i^S, y_i^S)\}, \tag{1}$$

where x_i^S – is the input data, y_i^S – is the corresponding output data (labels) of the dataset D_S, $i=1,...,N_S$, N_S – is the size of the dataset D_S.

Let us denote the basic model as a neural network trained on the data of the problem S with parameters θ_S as $f_{\theta S}$.

Let T be a target task for which there is a limited amount of labeled data (dataset D_T):

$$D_T = \{(x_j^T, y_j^T)\}, \tag{2}$$

where x_j^T – is the input data, y_j^T – is the corresponding output data (labels) of the dataset D_T, $j=1,...,N_T$, N_T is the size of the dataset D_T.

At the same time, using the parameters θ_S of the base model $f_{\theta S}$ to initialize the weights of the target model $f_{\theta T}$, it is possible to train the target model $f_{\theta T}$ on the data of the dataset D_T to adapt it to the target task.

A model $f_{\theta S}$ is called a pre-trained model if the minimum loss function is ensured for the target model $f_{\theta T}$ trained on the basis of parameters θ_S:

$$\theta_T{}^* = arg\ min_{\theta T}\, L_T(f_{\theta T}(x_j^T), y_j^T), \tag{3}$$

where L_T – is the loss function adopted for the target problem.

At the same time, to assess the quality of the pre-trained model $f_{\theta S}$ (measuring the success of the base model), we can introduce the concept of performance $P_{\theta T}$ of the model $f_{\theta S}$ on the target dataset D_T as a metric that characterizes the model's ability to solve the target problem:

$$P_{\theta T} = E_{\theta T} / t_{\theta T}, \tag{4}$$

where $E_{\theta T}$ – is the difference between the predicted $f_{\theta T}(x_j^T)$ and the true y_j^T values of the objective function; $t_{\theta T}$ – is the time spent on training the model with an accuracy of $E_{\theta T}$.

In practice, the mean absolute error *mae*, mean squared error *mse,* or coefficient of determination *R-squared can be* used as $E_{\theta T}$. The number of epochs of model training can be used as $t_{\theta T}$. Calculating the model performance allows you to assess how well the pre-trained model fits a specific target task and data.

The main part. The basic idea of pre-training is that a neural network is first trained to extract general features and patterns from data that can be applied to different tasks. This pre-trained model is then retrained on a narrower sample of data related to a specific task.

However, when implementing this method, the problem arises of finding the moment when pre-training stops, when the model is already able to extract general patterns from the underlying dataset and, at the same time, has not adapted to the data of a particular task, i.e., the neural network has not been retrained. Violation of this balance causes the following problems.

1. Early termination of pre-training (undertraining): if the pre-trained and target models differ significantly, the retraining process may take longer and be less effective.

2. Late termination of pre-training (retraining): if the pre-trained model has adapted to the data of the base dataset, then the problem of Domain Shift arises. This can lead to a loss of model performance on the target task due to the mismatch between the characteristics of the base and target datasets.

The developed method of pre-training neural networks should take into account both limitations by determining the pre-training threshold, which reduces the time of model training while ensuring a given accuracy.

A time-delay neural network (TDNN) is used as a neural network model of nonlinear dynamic objects with continuous characteristics. Due to their simplicity and versatility, TDNNs have become the most widespread. The most commonly used TDNN structure consists of three layers: input, hidden, and output [14].

In this structure, the input layer of a TDNN includes M neurons – the memory length of the object model. The number of neurons M choses to best reflect the dynamic properties of the object. The hidden layer includes K neurons with a nonlinear activation function. The number of neurons K choses to best reflect the nonlinear properties of the object. The output layer of TDNN contains 1 neuron with a linear activation function.

The input layer receives data

$$\mathbf{x}(t_n)=[x(t_n), x(t_{n-1}), \dots, x(t_{n-M-1})],\ t_n = n\Delta t,\ n=1, 2, \dots \tag{5}$$

The signal $y(t_n)$ at the output layer at time t_n depends on the values of the input signal $\mathbf{x}(t_n)$ and is determined by the expression [14]:

$$y(t_n) = b_0 + S_0 \sum_{i=1}^{K} w_i\, S_i \left(b_i + \sum_{j=1}^{M} w_{i,j}\, x(t_{n-j}) \right) \tag{6}$$

where b_0, b_i – are the biases of the neurons of the output and hidden layers, respectively; S_0, S_i – are the activation functions of the neurons of the output and hidden layers, respectively; w_i, $w_{i,j}$ – are the weighting coefficients of the neurons of the output and hidden layers, respectively.

Taking into account the well-known fact that in multilayer neural networks the first layer identifies the most general features, and the subsequent layers identify more specific features, the following method of pre-training neural networks with time delays is proposed in this paper for identifying nonlinear dynamic objects with continuous characteristics.

Stage 1. The neural network model f_S is trained on the data of the basic dataset $\{(\mathbf{x}_S(t_n), y_S(t_n))\}$. In this case, the criterion for stopping the training process is the simultaneous fulfillment of the following conditions: the standard deviation of the parameters of the input layer of the network at epochs $k+1$ and k does not exceed a predetermined value E_1 and the standard deviation of the parameters of the hidden layer of the network at epochs $k+1$ and k *is* not less than a predetermined value E_2:

$$\begin{cases} e_1 = \dfrac{1}{KM}\sum_{i=1}^{K}\sum_{j=1}^{M}\left(w_{i,j}^{k+1} - w_{i,j}^{k+1}\right)^2 \le E_1 \\ e_2 = \dfrac{1}{K}\sum_{i=1}^{K}\left(w_i^{k+1} - w_i^{k+1}\right)^2 \ge E_2 \end{cases} \tag{7}$$

Stage 2. The neural network model $y_T(t_n)$ is trained on the data of the target dataset $\mathbf{x}_T(t_n)$. The parameters of the base model $y_S(t_n)$ are used as the initial state of the target model $y_T(t_n)$. In the process of training the target model, the pre-trained weights of the neural network are not fixed in order to correct and adjust them to the data of the target task during further training. This process is called fine-tuning.

The criterion for stopping the learning process is the absolute deviation of the model output from the target values.

The proposed method of pre-training neural networks with time delays for the identification of nonlinear dynamic objects is tested on the task of identifying a test object with continuous characteristics.

Experimental setup. The accuracy of TDNNs and the time of their construction using pre-training of neural network models is studied on the example of a basic dataset. The dataset is formed from a set of input signals $x(t)=a\Theta(t)$ in the form of step functions with different amplitudes a and their corresponding output signals $y(t)=f(x(t))$.

In [15], it was established that TDNN models are not invariant to the shape of the input signal and can adequately reflect the properties of a dynamic object when trained on a sufficient amount of input and output signals of the same type as in the test data set. Therefore, the common nonlinear and dynamic links shown in Table 1 are used as a black box transducer f.

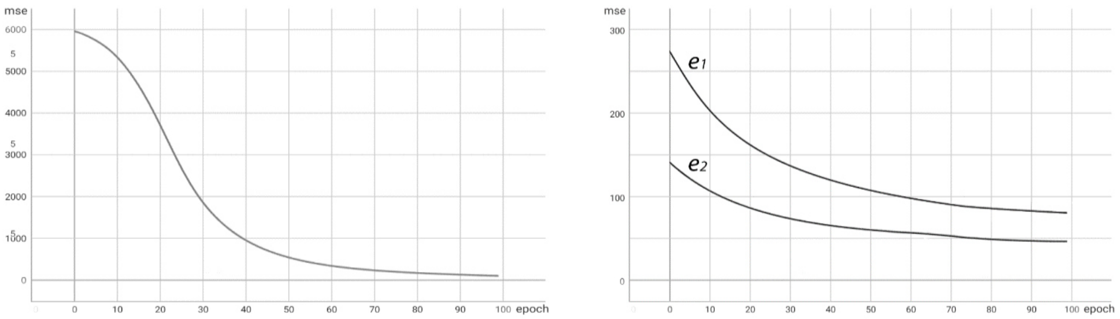
Table 1

Nonlinear and dynamic functions that form the basic dataset

№	Title	Expression
1	Inertia-free amplifier	$y(t)=Rx(t)$
2	Integrator	$y(t)=1/T \int_0^t x(t)d(t)$
3	Inertial link	$Tdy(t)/dt+ y(t) = x(t)$
4	Oscillating link	$c\, d_1{}^2\, y(t)/dt^2 + c_2\, dy(t)/dt + c_3\, y(t)= x(t)$
5	The lagging link	$y(t)=Kx(t-\tau),\ t>\tau$

A three-layer neural network with time delays is used as a pre-trained neural network model. In this structure, the input and hidden layers of the network have $M=K=50$ neurons with activation in the form of a linear rectifier [15]. The output layer of the TDNN includes 1 neuron with a linear activation function. This structure of the neural network ensures the level of losses set by the experimental conditions (*mse* < 50) with an acceptable training time (*epochs* < 100). The resulting TDNN was used to study the accuracy of models of dynamic objects with smooth and significant nonlinearities.

The process of preliminary training of the described model on the data of the basic dataset is demonstrated in Fig. 1: a plot of the loss function versus the number of training epochs (Fig. *1a)* and a plot of the model training accuracy versus the number of training epochs (Fig. *1b)*.



a *b*

Fig. 1. TDNN model training process on the base dataset: *a* – graph of the loss function versus the number of training epochs; *b* – graph of the model training accuracy metric versus the number of training epochs.

Taking into account the stopping conditions (7), the pre-learning process stops after the end of epoch **7**.

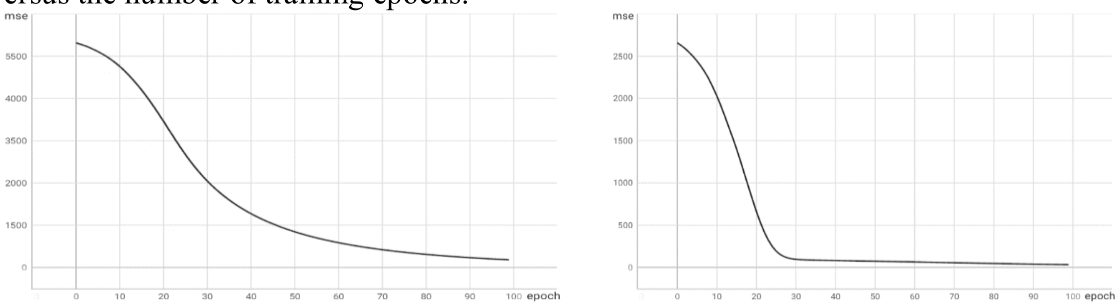
We study the accuracy of TDNNs and the time of their construction at the stage of target training using the example of a target dataset. The dataset is formed from a set of input signals $x(t)=a\Theta(t)$ in the form of step functions with different amplitudes a ($a\in(0,1)$) and the corresponding output signals [14] in the form of a linear function with saturation:

$$y(t)=\begin{cases} s, x(t)>p \\ gx(t), x(t)\le p \end{cases} \tag{8}$$

where s – is the saturation level, p is the saturation point, and g is the gain.

The process of training a TDNN model with randomly selected weights on the target dataset is shown in Fig. 2*a*: a graph of the model training accuracy metric versus the number of training epochs.

The process of training a TDNN model by retraining a pre-trained model on the target dataset is demonstrated in Fig. 2*b*: a plot of the model training accuracy metric versus the number of training epochs.



a *b*

Fig. 2. Graph of the model training accuracy metric versus the number of training epochs on the target dataset: *a* – models with randomly selected weights; *b* – by retraining a pre-trained model.

Fig. 2 shows a 3.7-fold reduction in the time required to train the TDNN model on the target dataset compared to the full training procedure, while ensuring comparable accuracy of both models.

To investigate the modeling accuracy, the resulting neural network is verified on a test nonlinear dynamic object given in [15]. In Fig. 3 shows a comparison of the output signals $y_n(t)$, $y_v(t)$ and $y(t)$, obtained as a result of the action of the step signal $x(t)=a\Theta(t)$ (a=0.65) at the inputs of the TDNN model, the second-order integral-step series [14] (chosen as a method of deterministic identification for comparison), and the simulation model of the test nonlinear dynamic object, respectively. The graph shows a 15% advantage in accuracy of the proposed TDNN model over the integral-step model.

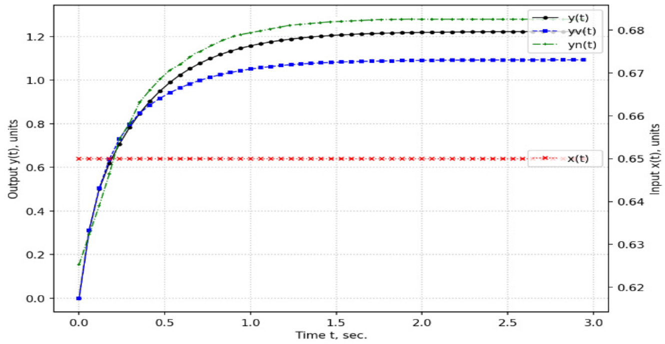


Fig. 3. Comparison of the output signals yn(t), yv(t) and y(t), obtained as a result of the action of the signal x(t)=aΘ(t) on the inputs of the TDNN model, the integro-power series and the simulation model of a nonlinear dynamic object, respectively .

Discussion of results. The obtained simulation results show that the use of TDNN models with pre-training to identify nonlinear dynamic objects with continuous characteristics can significantly reduce the training time of neural network models without losing accuracy. However, this method is demanding to meet the following conditions:

Task Mismatch: if the tasks for which the model has been trained and the target task differ significantly, the trained model may be less effective. A special case is the Domain Shift problem, where a pre-trained model is trained on data that is significantly different in distribution from your target data.

Insufficient Data for Fine-Tuning: if there is a limited amount of labeled data for the target task, then retraining a trained model may face the problem of overtraining or undertraining.

Model Size: some pre-trained models can be large and require large computing resources to run and retrain.

Thus, the area of effective application of the method of pre-training neural networks with time delays in the tasks of identifying nonlinear dynamic objects with continuous characteristics is when the general and target datasets do not have significant differences and the target dataset is of sufficient size to reflect the properties

Conclusions. In this paper, an attempt has been made to further develop the method of pre-training neural networks with time delays in the tasks of identifying nonlinear dynamic objects with continuous characteristics in order to reduce the training time of neural network models without significant loss of accuracy.

The novelty of the proposed method lies in the use of a formal criterion for determining the moment of termination of pre-training, which allows avoiding retraining of the base model and providing a significant reduction in the model training time on the target dataset.

The proposed method of pre-training neural networks with time delays for the identification of nonlinear dynamic objects is tested, which demonstrated a 3.7-fold reduction in model training time on the target dataset compared to the full training procedure while ensuring comparable accuracy of both models.

The advantages of pre-training include the ability to improve model performance when there is a lack of labeled data for the target task. This is especially useful in situations where the collection of labeled data requires a lot of effort and the base task has an excess of data. The area of effective use of the proposed method is highlighted.

References

1. Rudin C., Radin J. Why are we using black box models in AI when we don't need to? A lesson from an explainable AI competition. *Harvard Data Science Review*. 2019. V.2. No. 1.
2. Mitrea C.A., Lee C.K.M., Wu Z. A comparison between neural networks and traditional forecasting methods: A case study. *International journal of engineering business management*. 2009. V.1. No. 2. P. 19-24.
3. Karim R., Shajalal, Grass A. Interpreting Black-box Machine Learning Models for High Dimensional Datasets. *arXiv preprint*. 2022. URL: arxiv.org/abs/2208.13405.
4. Sen J. Machine Learning - Algorithms, Models and Applications. London, UK: IntechOpen, 2021.
5. Sharma S., Sharma S., Athaiya A. Activation functions in neural networks. International *Journal of Engineering Applied Sciences and Technology*. 2020. V.4, No 12. P. 310-316.
6. Karampiperis P., Manouselis N., Trafalis T.B. Architecture selection for neural networks. *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02*, *Honolulu, HI, USA*. 2002. V.2. P. 1115-1119.
7. Pasechnyuk D., Prazdnichnykh A., Evtikhiev M., Bryksin T. Judging Adam: Studying the Performance of Optimization Methods on ML4SE Tasks. *arXiv:2303.03540*
8. Hoefler T., Alistarh D., TalBen-Nun N.D., Peste A. Sparsity in Deep Learning: Pruning and growth for efficient inference and training in neural networks. *Journal of Machine Learning Research.* 2021. No 23. P.1-124.
9. Elsken T., Metzen J.H., Hutter F. Neural Architecture Search. In: Automated Machine Learning. The Springer Series on Challenges in Machine Learning. Cham: Springer, 2019. 223 p.
10. Li Z., Li H., Meng L. Model Compression for Deep Neural Networks: A Survey. *Computers.* 2023. No. 12. P.60.
11. Krykun V. Improving the accuracy of the neural network models interpretation of nonlinear dynamic objects. *Mathematical and computer modeling. Series: Technical sciences*. 2022. No. 23. P. 31-41.
12. Lu Y., Jiang X., Fang Y., Shi Ch. Learning to Pre-train Graph Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2021. V.33. No.5. 10p.
13. Hosna A., Merry E., Gyalmo J. Transfer learning: a friendly introduction. *J Big Data.* 2022. No. *9*. V.102.
14. Fomin O., Polozhaenko S., Krykun V., Orlov A., Lys D. Interpretation of Dynamic Models Based on Neural Networks in the Form of Integral-Power. *Lecture Notes in Networks and Systems*. 2022. V. 536. P. 258-265.
15. Fomin O., Speranskyy V., Krykun V., Tataryn O., Litynskyi V. Models of dynamic objects with significant nonlinearity based on time-delay neural networks. *Bulletin of Cherkasy State Technological University. Technical sciences.* 2023. No. 3. P. 97–112.

ВИКОРИСТАННЯ ПЕРЕДНАВЧЕНИХ НЕЙРОННИХ МЕРЕЖ ДЛЯ МОДЕЛЮВАННЯ НЕЛІНІЙНИХ ДИНАМІЧНИХ ОБ'ЄКТІВ

А.А. Орлов

Національний університет «Одеська політехніка»
1 Шевченка пр., Одеса, 65044, Україна
email: andrey.orlov.od@gmail.com

Розглянуто клас задач ідентифікації нелінійних динамічних об'єктів із безперервними характеристиками за допомогою нейронних мереж із часовими затримками. Обґрунтовано багаторазове використання попередньо навчених нейронних мереж для ідентифікації об'єктів різної природи, що мають схожі закони функціонування. Метою роботи є скорочення часу навчання нейромережевих моделей без значної втрати точності шляхом розвитку методу попереднього навчання нейронних мереж із часовими затримками в задачах ідентифікації нелінійних динамічних об'єктів із безперервними характеристиками. Наукова новизна роботи полягає у подальшого розвитку метод попереднього навчання нейронних мереж із часовими затримками в задачах ідентифікації нелінійних динамічних об'єктів з безперервними характеристиками, що дозволяє скоротити час навчання нейромережевих моделей без значної втрати точності. Метод полягає у вилученні загальних закономірностей із базового датасету на етапі попереднього навчання та використанні їх для розв'язання конкретних задач на етапі донавчання моделей на цільовому датасеті. Для визначення моменту припинення попереднього навчання нейронної мережі запропоновано формальний критерій, використання якого дає змогу уникнути перенавчання базової моделі та забезпечити суттєве скорочення часу навчання моделі на цільовому датасеті. Практичне значення роботи полягає в розробці алгоритму методу попереднього навчання нейронних мереж із часовими затримками в задачах ідентифікації нелінійних динамічних об'єктів з безперервними характеристиками, що дозволяє скоротити час навчання нейромережевих моделей втрати точності моделі. Дослідження швидкості збіжності алгоритму навчання та точності моделювання проведено експеримент з тестовими нелінійними динамічними об’єктами. Отримані результати моделювання свідчать про ефективність запропонованого методу. Цінність проведеного дослідження полягає у визначенні області ефективного використання запропонованого методу, а саме коли загальний та цільовий датасети не мають суттєвих розбіжностей та цільовий датасет має достатній розмір для відображення властивостей об’єкту дослідження.

Ключові слова: нелінійні динамічні об'єкти, моделювання, нейронні мережі з часовими затримками, переднавчання.