DOI: https://doi.org/10.15276/aait.07.2024.2
UDC 004.942

Modeling nonlinear dynamic objects using pre-trained time delay neural networks

Oleksandr O. Fomin[1)]
ORCID: https://orcid.org/0000-0002-8816-0652; fomin@op.edu.ua. Scopus Author Author ID: 57103429400
Andrii A. Orlov[1)]
ORCID: https://orcid.org/0000-0002-3256-5044; 9901020@stud.op.edu.ua. Scopus Author ID: 57202287340
[1)] Odessa Polytechnic National University, 1, Shevchenko Ave. Odessa, 65044, Ukraine

ABSTRACT

The work is devoted to resolving the contradiction between the accuracy of modelling nonlinear dynamics and the speed of model construction under conditions of limited computing resources. The purpose of the work is to reduce the time for building time delay neural networks while ensuring a given accuracy in the tasks of identifying nonlinear dynamic objects with continuous characteristics. This goal is achieved by developing a method for pre-training neural networks that reflect the basic characteristics of the subject area. The scientific novelty of the work is the development of a method for identifying nonlinear dynamic objects in the form of time delay neural networks based on the use of a set of basic pre-trained neural networks that reflect the typical properties of the subject area. In contrast to the traditional approach to pre-training, the developed method allows building models of lower complexity. A formal criterion is proposed for determining 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 data set. The practical utility of the work lies in the development of an algorithm for the method of pre-training time delay neural networks in the tasks of identifying nonlinear dynamic objects with continuous characteristics, which allows to significantly reduce the training time of neural networks without losing model accuracy. 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; time delay neural networks; pre-training

***For citation*: Fomin O. O., Orlov A. A. “Modeling nonlinear dynamic objects using pre-trained time delay neural networks”. *Applied Aspects of Information Technology.* 2024; Vol. 7 No.1: 24–33. DOI: https://doi.org/10.15276/aait.07.2024.2**

INTRODUCTION

In today's fast-paced world, technologies play a key role in the progress of most aspects of human activity. From biological and medical research, industrial and energy processes to financial and logistics systems, technological innovations are opening up new horizons of opportunity [1, 2].

Successes in the development of technologies are due, in particular, to the complexity of control objects, which are reflected in the gradual change of linear models to more complex non-linear ones, in particular, those with dynamic properties.

Such models are the basis of most objects and processes of the surrounding world and have a number of advantages: increasing the accuracy of modeling, expanding the range of external conditions and modes of operation. This paves the way for more reliable and efficient solutions that can adapt to changing conditions and requirements.

However, along with the advantages, nonlinear dynamic models put forward increased requirements for the accuracy and flexibility of identification systems.

A common approach to modeling nonlinear dynamics are neural networks of various architectures, in particular, time delay neural networks [3]. One of the key advantages of neural networks is their ability to build models based on experimental input/output data, making them exceptionally useful for solving problems where complex relationships between variables cannot be easily understood or described using traditional mathematical models [4]. This property is especially valuable in situations where big data about the behavior of the system is available, but there is a lack of a priori information about the structure and internal properties of the object.

Among the disadvantages of using neural networks in modeling are the complexity of interpreting the resulting models and a significant training time [4, 5].

Thus, there is a contradiction between the accuracy of modeling nonlinear dynamics and the speed of model construction under conditions of limited computing resources or when a rapid response is required.

In the current situation, a pre-training approach can serve as an effective means of overcoming the established contradiction, as it allows you to start training the model already with some understanding of the data, which speeds up the learning process.

Based on the analysis of the advantages and disadvantages of traditional methods of preliminary learning, an attempt is made to eliminate the contradiction between the accuracy of modeling nonlinear dynamics and the speed of model construction by developing and adapting methods of preliminary training of time delay neural networks.

LITERATURE REVIEW

To solve the contradiction that hinders the development of neural networks in the direction of identifying nonlinear dynamics, several approaches have been proposed depending on the properties of the object, the amount of a priori data, and the conditions of application [6]. Analysis and systematization of these approaches to the development of methods for accelerating the training of neural networks and rapid adaptation to new tasks on the basis of limited data allows us to identify several areas [1, 7]. Among them: improvement of activation functions [5, 8], teaching methods [6, 9] and the structure of neural networks [7].

In recent years, an approach based on pre-training of neural networks has become popular due to its effectiveness [10, 11]. Pre-training of neural networks consists of extracting information from previous data before the start of the main stage of training. This makes it possible to start training the model on target data using a pre-trained model instead of a model with random parameters, which allows you to accelerate the convergence of the neural network model, increases its performance even on datasets of limited volume [11, 12].

The implementation of the pre-training technique is carried out in two stages [10, 13]:

– *construction of a rough model*: extraction of general patterns from the basic dataset by means of preliminary training of models;

– *Building an exact model*: training a rough model on the target dataset.

At the same time, once trained, a rough model characterized by a simplified representation of complex objects can be used in modeling any objects whose laws of functioning have similar features.

The described approach has several predominant features at once [10, 11].

First, the use of pre-trained models reduces training time on target data. As a result, it significantly reduces the computational cost of training an accurate model.

Secondly, it makes it possible to extract general patterns from large amounts of data, which is especially useful when there is a lack of sufficient data in the target dataset.

The disadvantages of the approach are the difficulties of adapting to tasks that are very different from those on which they were previously trained. In addition, pre-trained models can be prohibitively complex to model target objects.

There are several approaches to pre-training neural networks [11, 14].

Autocoding. This is neural network training, during which the model tries to reproduce the input data at the output, minimizing the loss between the original and recovered data.

Deep learning on unlabeled data. This method involves the use of neural networks to learn from large amounts of unlabeled data.

Transfer Learning. In the transfer of learning, the model is pre-trained on one task and then continues to learn on another related task.

These pre-training methods can be used individually or in combination with each other, depending on the specific task and available data.

Pre-trained neural networks have become an integral part of much research and development in the field of artificial intelligence [15]. They have proven their effectiveness in reducing the training time of convolutional neural networks and building universal models in a wide range of tasks. The pre-trained convolutional networks DenseNet, VGG, which are successfully used in computer vision tasks, the BERT natural language processing network, and the GPT text generation network are well known.

This direction also looks promising in the problems of identification of nonlinear dynamic objects. At the same time, there is a lack of work in the field of preliminary training of neural networks that model nonlinear dynamic properties of objects with continuous characteristics.

Based on the above, the current direction of development of neural network methods for identifying nonlinear dynamic objects is the use of preliminary training, which is dynamically developing and able to effectively cope with the requirements of modern modeling tasks.

PROBLEM STATEMENT

The formal formulation of the problem of pre-training a neural network is as follows.

Let S be a domain for which there is marked-up data of sufficient size N_S (dataset D_S):

$$D_S=\{(\mathbf{x}_i^S, y_i^S)\}, \quad (1)$$

where $\mathbf{x}_i^S$ is the vector of independent variables, y_i^S is the corresponding target variable (label), i=1,..., N_S.

Let $f_{\theta S}$ is a general (rough) model in the form of a neural network with θ_S parameters, trained on D_S data.

Let T is a target problem in domain S for which there is labeled data of limited size N_T (dataset D_T):

$$D_T=\{(\mathbf{x}_j^T, y_j^T)\}, \quad (2)$$

where $\mathbf{x}_j^T$ is the vector of independent variables; y_j^T is the corresponding target variable (label), j=1,..., N_T.

Let $f_{\theta T}$ is a target (exact) model in the form of a neural network with θ_T parameters, trained on D_T data, which provides accuracy [16]:

$$E_{\theta T} = mse(f_{\theta T}(x_j^T) - y_j^T), \quad (3)$$

with the duration of training of the target model $t_{\theta T}$.

It is necessary to find the following parameters θ_S of the rough model, using which the initial values of θ_{T0} when training the exact model $f_{\theta T}$ achieves a given level of accuracy $E_{\theta T}$ in a minimum period of time:

$$\theta_{T0} = \theta_S : \underset{t_{\theta T}}{arg\ min}\, L_T(f_{\theta T}(\mathbf{x}_j^T), y_j^T) = E_{\theta T}, \quad (4)$$

where L_T is the loss function adopted for the target model.

When condition (4) is met, *the* $f_{\theta S}$ model is said to be pre-trained.

To assess the quality of the pre-trained $f_{\theta S}$ model, the learning performance indicator $P_{\theta T}$ of the $f_{\theta T}$ model on the target DT dataset is used in the form of the following metric [17]:

$$P_{\theta T}=E_{\theta T}/t_{\theta T}. \quad (5)$$

The number of epochs of model training can be used as the value of $t_{\theta T}$.

PURPOSE AND OBJECTIVES OF THE STUDY

The purpose of the work is to reduce the time of construction of time delay neural networks while ensuring the specified accuracy in the tasks of identifying nonlinear dynamic objects with continuous characteristics by developing a method of preliminary training of neural network models.

To achieve the goal, the following tasks are established**.**

1. Development of a method for identifying nonlinear dynamic objects based on basic pre-trained neural networks.
2. Construction of a criterion for stopping previous learning to prevent adaptation to the data of the basic dataset.
3. To study the effectiveness of the use of basic pre-trained neural networks when working with nonlinear dynamic objects containing combined characteristics in the form of a composition of typical dynamic links.

IDENTIFICATION METHOD BASED ON BASIC PRE-TRAINED NEURAL NETWORKS

1. Modeling of nonlinear dynamic objects based on basic pre-trained neural networks

An approach to training neural networks based on the use of prior learning in practice can lead to a slight increase or, in general, to a decrease in the training performance of the target model [13].

One of the factors reducing the productivity of the target model training process is the general nature and large volume of the training dataset D_{S0}, which should contain a description of the object's behavior in the widest possible range of external conditions and under the influence of a wide range of input signals [18, 19]. As a result, in a particular case, when solving the problems of modeling objects in a narrower range of external conditions and input influences, the rough $f_{\theta S0}$ and exact $f_{\theta Ti}$ models (i=1,...,g, where g is the number of target modeling tasks) have excessive complexity (Fig. 1a).

There are two ways to overcome this problem.

1. Formation of a set of separate highly specialized training datasets D_{Si} for the construction of rough models $f_{\theta Si}$ with the subsequent construction of exact models $f_{\theta Ti}$ by further training of a rough model on the target dataset D_{Ti} (Fig. 1b). In this case, the benefits of pre-training are largely lost, since it is necessary to form a separate training dataset D_{Si} for each target task and teach the rough model $f_{\theta Si}$. This leads to a loss of time in solving each target problem.

2. Formation of a set of separate training datasets D_{Sj} (j=1,…, h, where h – *is the* number of basic characteristics reflecting individual properties of the subject area), for the construction of basic pre-trained neural networks $f_{\theta Sj}$ with the subsequent construction of exact $f_{\theta Ti}$ models by retraining a

 ISSN 2617-4316 (Print)
ISSN 2663-7723 (Online)

rough model in the form of a composition of basic pre-trained neural networks on the target dataset D_{Ti} (Fig. 1c).

The result of this approach is a set of basic pre-trained neural networks $f_{\theta Sj}$ that are defined only once. At the same time, each of the basic neural networks $f_{\theta Sj}$ is much simpler than the rough model $f_{\theta S0}$.

A rough model of the object $f_{\theta Si}$, containing combined characteristics in the form of a composition of typical dynamic links, is built on the basis of a set of basic pre-trained neural networks $f_{\theta Sj}$ that correspond to the existing characteristics of the object. At the same time, the structure of the rough model $f_{\theta Si}$ (dimension of the parameter vector θ_{Sj}) must coincide with the structure of the basic neural network $f_{\theta Sj}$ (dimension of the parameter vector θ_{Sj}):

$$f_{\theta Si} : dim(\theta_{Si}) = dim(\theta_{Sj}), \quad (6)$$

which ensures the simplicity of an exact model $f_{\theta Ti}$.

In this paper, a method for identifying nonlinear dynamic objects is proposed, based on the use of a set of basic pre-trained neural networks reflecting typical properties of the subject area.

The algorithm of the proposed method is to perform the following steps.

1. Selection of typical domain properties and formation of a set of datasets reflecting the selected properties.
2. Preliminary training of basic neural networks on the generated datasets that correspond to the established typical properties of the subject area.
3. Construction of a rough model based on a set of basic pre-trained neural networks that correspond to the properties of a certain target task.
4. Training of an accurate neural network model based on the rough model obtained at the previous stage.

2. Selection of typical domain properties for the formation of a set of datasets

The procedure for selecting typical domain properties for the formation of a set of datasets includes the following steps [20, 21], [22].

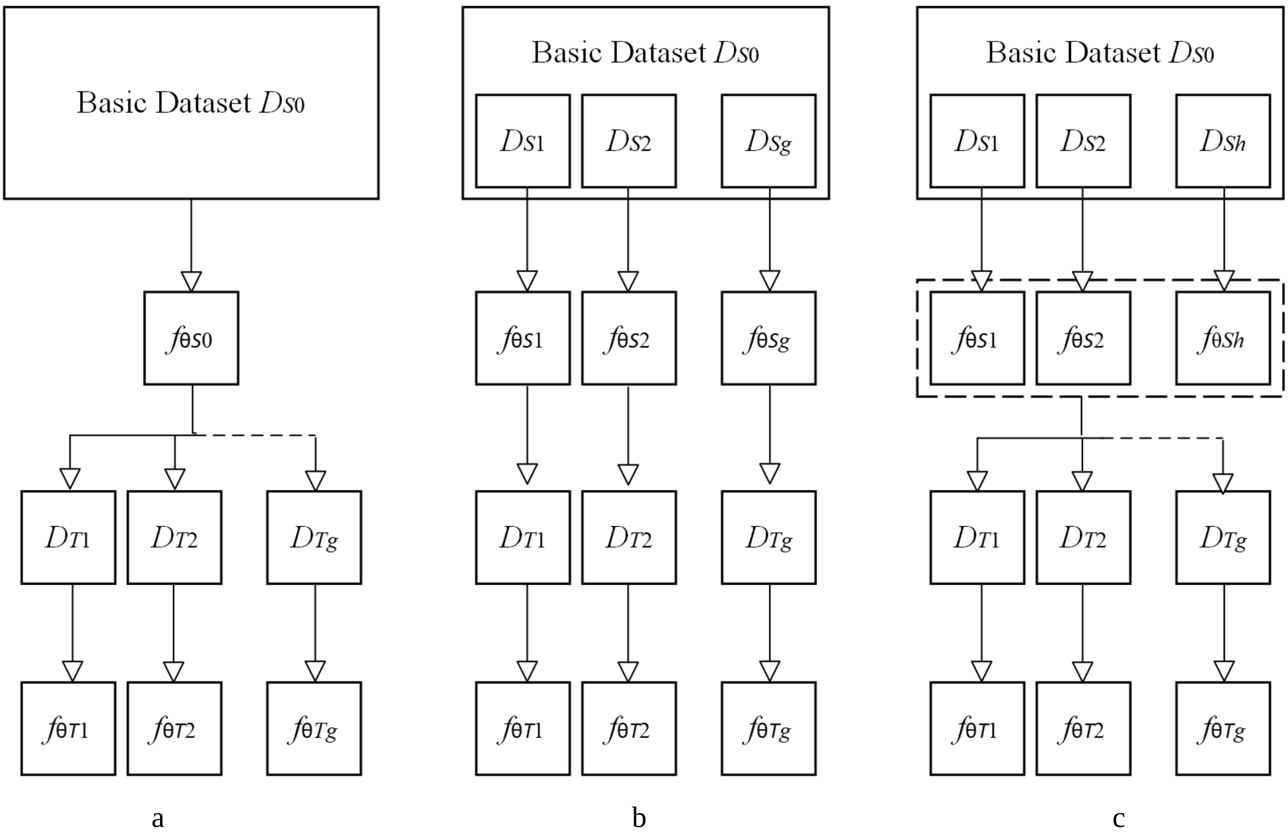

Fig. 1. **Block diagram of the pre-learning process:**
a – preliminary training of the basic model; b – pre-training of simplified models;
c – pre-training of basic models
Source: **compiled by the authors**

1. Determination of the range of tasks to be solved in the subject area, analysis of the properties of the subject area, which are essential for the objects of the subject area and should be reflected in the formation of the dataset D_{S0}.

2. Determination of the types of signals (e.g., periodic, random, pulsed) that best reflect the dynamics of the objects under study and the reaction of the object to which should be included in the dataset D_{S0}. Determination of signal parameters for each type of signal, such as amplitude, frequency, phase, pulse duration, etc., corresponding to the typical operating conditions of domain objects.

3. Formation of the dataset D_{S0} based on the list of domain properties and the generated set of input signals. Segmentation of the D_{S0} dataset into separate datasets D_{Sj} (j=1,…,h) in accordance with the defined list of domain properties.

This procedure is iterative and involves adjusting a set of typical domain properties and input signal types and corresponding datasets.

3. Pre-training of basic neural networks

The idea of pre-training is to extract general patterns and dependencies from a common dataset. The acquired knowledge can be used in solving various target problems in the subject area. To do this, the pre-trained model continues to be trained on the target dataset that corresponds to a specific target task.

However, when implementing this approach, the problem arises of finding the best moment to stop the process of preliminary training of the basic neural network. At this point, the pretrained model should already contain the general patterns from the basic dataset D_{Si} and, at the same time, should not be retrained (it has not adapted to the data of this dataset). Violation of this state of equilibrium leads to the following undesirable consequences.

1. *Early termination of studies*. In the case where the baseline and exact models differ significantly, the process of training the exact model can become longer and less efficient.

2. *Belated termination of training.* If the base model has adapted well to the training dataset, the domain shift problem is likely to occur. This problem leads to a decrease in the performance of training an exact model on the target dataset D_{Ti} (i=1,…,g) due to a mismatch between the characteristics of the basic and target datasets.

The developed method for identifying nonlinear dynamic objects, based on the use of a set of basic pre-trained neural networks, should take into account both limitations. The implementation of the above constraints is carried out by setting a limit on the training time of basic models while ensuring a given accuracy.

To determine the training time thresholds for basic models, consider a time delay neural network (TDNN) as a powerful and effective tool for identifying nonlinear dynamic objects with continuous characteristics. The most popular TDNN structure is a three-layer direct signal propagation network consisting of input, hidden, and output layers [22, 23]:

– the input layer includes M neurons (M is the memory length of the object model). The value of M is chosen in such a way as to best reflect the dynamic properties of the object;

– the hidden layer contains K neurons with a nonlinear activation function. The value of K is chosen to best reflect the nonlinear properties of the object;

– the output layer consists of 1 neuron with a linear activation function.

A block diagram of TDNN is shown in Fig. 2.

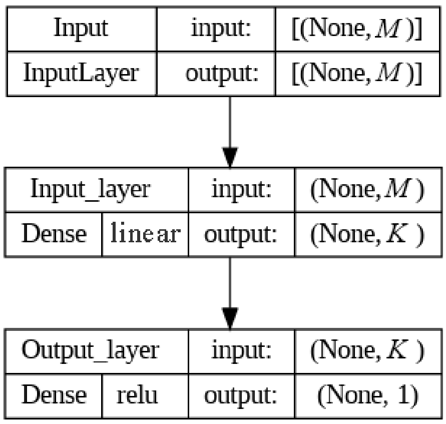

Fig. 2. **Time delay neural network block diagram with *M* inputs and *K* hidden neurons**
Source: **compiled by the authors**

The value of *None* in the data dimension vector in Fig. 2 means the variable number of rows in the dataset.

The input layer of the neural network receives the data:

$$\mathbf{x}_n^S=[x(t_n),\ x(t_{n-1}),\ \dots,\ x(t_{n-M-1})],\ t_n=n\Delta t,\ n=1,2,\dots \quad (7)$$

The signal y_n^S on the output layer at time t_n depends on the values of the input signal $\mathbf{x}_n^S$ and is determined by the expression [24]:

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

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

The criterion for stopping the learning process of the basic model $f_{\theta Si}$ (i=1,…,g) in the form of TDNN is the simultaneous fulfillment of two conditions [5, 8]:

– the standard deviation of the weights of the input layer of the network at the learning epochs $k+1$ and k does not exceed the specified value $E1$;

– the standard deviation of the weights of the hidden layer of the network at the learning epochs $k+1$ and k is not less than the specified value $E2$:

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

4. Building a rough model based on a set of basic pre-trained neural networks

After the pre-training process of the set of basic models is completed, a rough model is built on their basis. This model consists of a set of basic pretrained neural networks $f_{\theta Sj}$ that correspond to the existing typical characteristics of the object. Given the limitations (6) in the work, the parameters of the rough model θ_{Si} are determined from the composition of the corresponding parameters of the basic pretrained neural networks:

$$\theta_S = \frac{1}{d}\sum_{i=1}^{d}\theta_{Sd} , \quad (8)$$

where d is the number of basic pretrained neural networks corresponding to the typical characteristics of an object, $d \le h$.

At the same time, the dimension of the rough model $f_{\theta S}$ (dimension of the parameter vector θ_S) remains the same as that of the basic pre-learned models, that is, the complexity of the rough model does not increase.

Another advantage of forming a rough model from expression (8) is that there is no training procedure, which significantly speeds up the process of building a rough model.

5. Teaching an exact neural network model based on a rough model

Once a rough model is constructed, its θ_S parameters are taken as initial when training an exact model $f_{\theta Ti}$ (i=1, …,g): $\theta_{T0} = \theta_S$.

At the same time, the criterion for stopping the process of training an exact model is the standard deviation of the model's output from the values of the target variable (3).

Approbation of the developed method of identification of nonlinear dynamic objects, based on the use of a set of basic pretrained TDNN, is carried out on the task of modeling a test object with continuous characteristics [25].

EXPERIMENT SETUP

As a result of the work, the problem of developing a multifactor information criterion for choosing a machine learning model in the form of a neural network, which best meets the set of requirements for the accuracy and interpretability of an intelligent system, has been successfully solved.

The study of the effectiveness of the use of basic pre-trained neural networks in the identification of nonlinear dynamic objects is carried out on a test example.

The test object is a structure with a linear dynamic link and a nonlinear link in feedback [25, 26].

The selection of typical properties of the test object for the formation of a set of datasets reflecting the selected properties is carried out from typical dynamic links. The selected typical properties of the object are shown in Table 1.

Table 1. **Typical properties of the test object**

No.	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	$T_1 d^2y(t)/dt^2+ T_2\, dy(t)/dt + T_3 y(t)= x(t)$
5	Link with saturation	$y(t) = \begin{cases} a, x(t) > p \\ x(t), x(t) \le b \end{cases}$

***Source:* compiled by the authors**

The training sample obtained from the results of the simulation experiment "input/output" with the test object is a set of signals – responses of the object $y(t)$ to the test input signals $x(t)$ in the form of impulse, step, linear and harmonic functions with different amplitudes a.

Preliminary training of basic neural networks was carried out on the basis of three-layer TDNNs with the number of neurons in the input and hidden layers M=K=50. The level of loss (*mse* < 50) and the training time (*epochs* < 100) established by the conditions of the experiment.

The construction of a rough model is carried out for an object with the properties of an inertia-free amplifier, an inertial link and a saturation function on the basis of a set of corresponding basic pre-trained neural networks according to expression (8).

Training of the exact neural network model $f_{\theta T1}$ is carried out on the basis of the obtained rough model in the form of TDNN on the data of the target dataset D_{T1}. The process of further training in the form of dependence of the metric of model training accuracy depending on the number of epochs of training is shown in Fig. 3.

For comparison, an exact model $f_{\theta T2}$ was trained in the form of TDNN of a similar structure with random initial values of weights on the data of the same target dataset D_{T1}. The learning process in the form of dependence of the metric of model training accuracy depending on the number of epochs of training is shown in Fig. 3.

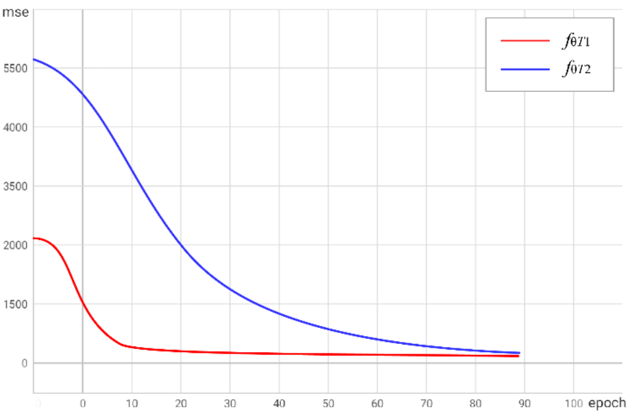

Fig. 3. **Graph of the $f_{\theta T2}$ *and* $f_{\theta T2}$ *model training accuracy metric as a function* of the number of epochs of training on the target dataset data D_{T1}**
Source: **compiled by the authors**

Fig. 3 shows the advantages of using basic pre-trained neural networks in the identification of nonlinear dynamic objects, which consist in reducing the training time of the TDNN model by 4.7 times compared to the full training procedure with comparable accuracy of both models.

DISCUSSION OF THE RESULTS

The obtained modeling results indicate that the use of TDNN models to identify nonlinear dynamic objects with continuous characteristics based on basic pre-trained neural networks can significantly reduce the training time of neural network models without loss of accuracy.

The advantages of the developed method of identification of nonlinear dynamic objects based on basic pre-trained neural networks include the ability to improve the performance of the model in the absence of labeled data for the target task. This is especially useful in situations where collecting labeled data requires a lot of effort and the baseline task has an overabundance of data. The area of effective use of the proposed method has been allocated.

The use of the developed method has limitations in its use.

Firstly, the common and target datasets should not differ significantly in the distribution of parameters in order to prevent the domain shift problem.

Secondly, insufficient data for fine-tuning can lead to the problem of overtraining or insufficient training of the model.

Thus, the sphere of effective application of the method of identification of nonlinear dynamic objects with continuous characteristics based on the use of a set of basic pre-trained neural networks is the cases 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 object of study.

It should be noted that the paper does not consider the procedure for selecting typical properties of the subject area and signals that best reflect the dynamics of the objects under study.

CONCLUSIONS

As a result of the work, the problem of reducing the time for building time delay neural networks while ensuring the specified accuracy in the tasks of identifying nonlinear dynamic objects with continuous characteristics is successfully solved by using a set of basic pre-trained neural networks that reflect the typical properties of the subject area.

To resolve the contradiction between the accuracy of modeling nonlinear dynamics and the speed of model construction, a method for identifying nonlinear dynamic objects based on basic pre-trained neural networks has been developed.

To prevent early or delayed termination of training of a rough model, a criterion for stopping training based on the assessment of the standard deviation of the parameters of the input and hidden layers of the network for models in the form of

 ISSN 2617-4316 (Print)
ISSN 2663-7723 (Online)

TDNN is proposed.

The efficiency of the developed method for identifying nonlinear dynamic objects based on the use of a set of basic pre-trained neural networks has been proven in solving the problem of identifying a test nonlinear dynamic object, which allows to reduce the training time of the TDNN model by 4.7 times compared to the full training procedure with comparable accuracy of both models.

REFERENCES

1. Hamm, P. & Klesel, M. "Success factors for the adoption of artificial intelligence in organizations: A literature review". *AMCIS Proceedings*. 2021.

2. Wirtz, B. W., Weyerer, J. C. & Geyer, C. "Artificial intelligence and the public sector – applications and challenges". *International Journal of Public Administration.* 2019; 42 (7): 596–615, https://www.scopus.com/record/display.uri?eid=2-s2.0-85050557949&origin=resultslist. DOI: https://doi.org/10.1080/01900692.2018.1498103.

3. Elsken, T., Metzen, J. H. & Hutter, F. "Neural architecture search: A survey". In: Hutter, F., Kotthoff, L., Vanschoren, J. (eds). *Journal of Machine Learning Research*. *Springer:* Cham. 2019; 20, https://www.scopus.com/record/display.uri?eid=2-s2.0-85068557085&origin=resultslist. DOI: https://doi.org/10.1007/978-3-030-05318-5_3.

4. Mitrea, C. A., Lee, C. K. M. & Wu, Z. "A comparison between neural networks and traditional forecasting methods: case study". *International Journal of Engineering Business Management*, 2009; 1 (2): 19–24, https://www.scopus.com/record/display.uri?eid=2-s2.0-84997830620&origin=resultslist. DOI: https://doi.org/10.5772/6777.

5. Sen, J. "Machine learning – algorithms, models and applications". *IntechOpen*. London: United Kingdom. 2021. DOI: https://doi.org/10.5772/intechopen.94615.

6. Chisty, N. M. A. & Adusumalli, H. P. "Applications of artificial intelligence in quality assurance and assurance of productivity". *ABC Journal of Advanced Research.* 2022; 11 (1): 23–32. DOI: https://doi.org/10.18034/abcjar.v11i1.625.

7. 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; .2: 1115–1119. DOI: https://doi.org/10.1109/IJCNN.2002.1007650.

8. Kutyniok, G. "The mathematics of artificial intelligence". *Proc. Int. Cong.* Math. 2022; 7: 5118–5139. DOI: https://doi.org/10.4171/ICM2022/141.

9. Kästner, C. & Kang, E. "Teaching software engineering for AI-enabled systems". *The 42nd International Conference on Software Engineering (ICSE). Software Engineering Education and Training*. 2020. Available from: https://arxiv.org/abs/2001.06691. [Accessed: Dec 2023].

10. Hosna, A., Merry, E., Gyalmo, J. et al. "Transfer learning: a friendly introduction". *J Big Data*. 2022; 9 (1): 102, https://www.scopus.com/record/display.uri?eid=2-s2.0-85140596814&origin=resultslist. DOI: https://doi.org/10.1186/s40537-022-00652-w.

11. Kamath, U, Liu J. & Whitaker, J. "Transfer learning: Domain adaptation". *In: Deep Learning for NLP and Speech Recognition*. Springer; 2019. p. 495–535. DOI: DOI:10.1186/s40537-022-00652-w.

12. Arsirii, O. O. & Petrosiuk, D. V. "Pseudo-labeling of transfer learning convolutional neural network data for human facial emotion recognition". *Herald of Advanced Information Technology*. 2023; 6 (3): 203–214. DOI: https://doi.org/10.15276/hait.06.2023.13.

13. Tercan, H., Guajardo, A., Heinisch, J., Thiele, T., Hopmann, C. & Meisen, T. "Transfer-learning: bridging the gap between real and simulation data for machine learning in injection molding". *Procedia CIRP*. 2018; 72: 185–90. DOI: https://doi.org/10.1016/j.procir.2018.03.087.

14. Hussain, M., Bird, J. J. & Faria, D. R. "A study on CNN transfer learning for image classification". *In: UK Workshop on Computational Intelligence*. *Springer.* 2018. p. 191–202, https://www.scopus.com/record/display.uri?eid=2-s2.0-85052244260&origin=resultslist. DOI: https://doi.org/10.1007/978-3-319-97982-3_16.

15. Lu, Y., Jiang, X., Fang, Y. & Shi. Ch. "Learning to Pre-train Graph Neural Networks". *Proceedings of the AAAI Conference on Artificial Intelligence*. May 2021; 33 (5): 4276–4284. DOI: https://doi.org/10.1609/aaai.v35i5.16552.

16. Tian, J. "Quality-evaluation models and measurements". *IEEE Software*. May-June 2004; 21 (3): 84–91. DOI: https://doi.org/10.1109/MS.2004.1293078.

17. Trzęsiok, M. "Measuring the quality of multivariate statistical models". *Acta Universitatis Lodziensis. Folia Oeconomica*. 2019; 6 (339): 99–100. DOI: https://doi.org/10.18778/0208-6018.339.06.

18. Cutler, M., Walsh, T. J, How, J. P. "Real-world reinforcement learning via multifidelity simulators". *IEEE Trans Robot.* 2015; 31 (3): 655-671.

19. Kumar, R, Jha, P. & Sahula, V. "An augmented translation technique for low resource language pair: Sanskrit to hindi translation". In: *Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence*, 2019. p. 377–383.

20. Pal, K. & Patel, B. V. "Data classification with K-fold cross validation and holdout accuracy estimation methods with 5 different machine learning techniques". *Fourth International Conference on Computing Methodologies and Communication (ICCMC)*. Erode: India. 2020. p. 83–87. DOI: https://doi.org/10.1109/ICCMC48092.2020.ICCMC-00016.

21. Ghallab, M. "Responsible AI: requirements and challenges". *AI Perspect*. 2019; 1 (3): 1–7. DOI: https://doi.org/10.1186/s42467-019-0003-z.

22. Tsmots, I. G., Berezsky, O. M. & Berezkyy, M. O. "Methods and hardware to accelerate the work of a convolutional neural network". *Applied Aspects of Information Technology*. 2023; 6 (1): 13–27. DOI: https://doi.org/10.15276/aait.06.2023.1.

23. Nakamichi, K. et al. "Requirements-driven method to determine quality characteristics and measurements for machine learning software and it's evaluation". *IEEE 28th International Requirements Engineering Conference (RE)*. Zurich: Switzerland. 2020. p. 260–270. DOI: https://doi.org/10.1109/RE48521.2020.00036.

24. Wenyuan, L., Zhu, L., Feng, F. at al. "A time delay neural network based technique for nonlinear microwave device modelling". *Micromachines*. Basel, 2020; 11 (9): 1–14.

25. Fomin, O. et al. "Interpretation of dynamic models based on neural networks in the form of integral-power series". In: Arsenyeva, O., Romanova, T., Sukhonos, M., Tsegelnyk, Y. (eds). *Smart Technologies in Urban Engineering. Lecture Notes in Networks and Systems*. *Springer*. Cham. 2022; 536: 258–265, https://www.scopus.com/record/display.uri?eid=2-s2.0-85050654902&origin=resultslist. DOI: https://doi.org/10.1007/978-3-031-20141-7_24.

26. 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; 3: 97–112. DOI: https://doi.org/10.24025/2306-4412.3.2023.288284.

Conflicts of Interest: the authors declare no conflict of interest

Received 09.01.2024
Received after revision 18.03.2024
Accepted 22.03.2024

DOI: https://doi.org/10.15276/aait.07.2024.2
УДК 004.942

Моделювання нелінійних динамічних об'єктів із використанням попередньо навчених нейронних мереж із часовими затримками

Фомін Олександр Олексійович[1)]
ORCID: https://orcid.org/0000-0002-8816-0652; fomin@op.edu.ua. Scopus Author ID: 57103429400
Орлов Андрій Андрійович[1)]
ORCID: https://orcid.org/0000-0002-3256-5044; 9901020@stud.op.edu.ua. Scopus Author ID: 57202287340
[1)] Національний університет «Одеська політехніка», проспект Шевченка, 1. Одеса, 65044, Україна

АНОТАЦІЯ

Робота присвячена вирішенню протиріччя між точністю моделювання нелінійної динаміки і швидкістю побудови моделі в умовах обмежених обчислювальних ресурсів. Метою роботи є скорочення часу побудови нейронних мереж з часовими затримками при забезпеченні заданої точності в задачах ідентифікації нелінійних динамічних об'єктів із безперервними характеристиками. Ця мета досягається шляхом розробки методу попереднього навчання нейронних мереж, що відображають базисні характеристики предметної області. Наукова новизна роботи полягає у розробці методу ідентифікації нелінійних динамічних об'єктів у вигляді нейронних мереж з часовими затримками на основі використання набору базисних попередньо навчених нейронних мереж, що відображають типові властивості предметної області. На відміну від традиційного підходу до попереднього навчання, розроблений метод дозволяє будувати моделі меншої складності. 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 наборі даних. Практична користь 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 суттєво скоротити час навчання нейронних мереж без втрати точності моделі. Цінність проведеного дослідження полягає у визначенні області ефективного використання запропонованого методу, а саме коли загальний та цільовий набір даних не мають суттєвих розбіжностей та цільовий набір даних має достатній розмір для відображення властивостей об’єкту дослідження.

Ключові слова: нелінійні динамічні об'єкти; моделювання; нейронні мережі з часовими затримками; попереднє навчання

ABOUT THE AUTHORS

Oleksandr O. Fomin - Doctor of Engineering Sciences, Professor. Department of Computerized Systems and Software Technologies. Odessa Polytechnic National University, 1, Shevchenko Ave. Odessa, 65044, Ukraine
ORCID: https://orcid.org/0000-0002-8816-0652; fomin@op.edu.ua. Scopus Author ID: 57103429400
***Research field*:** Math modeling; information systems; intelligent control systems

Фомін Олександр Олексійович - доктор технічних наук, професор кафедри Комп’ютеризованих систем та програмних технологій. Національний університет «Одеська політехніка», проспект Шевченка, 1, Одеса, Україна

Andrii A. Orlov – graduate student, Department of Computerized Systems and Software Technologies, Odessa Polytechnic National University, 1, Shevchenko Av. Odessa, Ukraine
ORCID: https://orcid.org/0000-0002-3256-5044; 9901020@stud.op.edu.ua. Scopus Author ID: 57202287340
Research field: Information technologies; artificial neural networks

Орлов Андрій Андрійович – аспірант кафедри Комп’ютеризованих систем та програмних технологій, Національний університет «Одеська політехніка», проспект Шевченка, 1, Одеса, Україна