

Investigation of the effectiveness of the Hybrid System of Computational Intelligence based on bagging and Group Method of Data Handling in the task of stock index forecasting^{*}

Yevgeniy Bodyanskiy^{1,†}, Yuriy Zaychenko^{2,†}, Oleksii Kuzmenko^{2,*,†} and Helen Zaichenko^{2,†}

¹ Artificial Intelligence Department, Kharkiv National University of Radio Electronics, Nauky Avenue 14, Kharkiv, 61166, Ukraine

² Institute for Applied System Analysis, Igor Sikorsky Kyiv Polytechnic Institute, Beresteiskyi Avenue 37-A, Kyiv, 03056, Ukraine

Abstract

The article considers intelligent methods of different generations for solving the problem of short- and medium-term forecasting of stock indices. The effectiveness of a fully connected network (BackPropagation), Group Method of Data Handling (GMDH), and Hybrid System of Computational Intelligence based on bagging and Group Method of Data Handling (HSCI-bagging) is considered. The influence of experimental parameters on the MSE and MAPE forecasting quality criteria is investigated. The obtained results are analyzed and the optimal parameters for different forecasting intervals are determined. On the basis of the optimal parameters found, the expediency of using the hybrid HSCI-bagging system in the forecasting task at different intervals is substantiated.

Keywords

Hybrid System of Computational Intelligence, bagging, GMDH, BackPropagation, forecasting

1. Introduction

The progress of intelligent methods has evolved in several stages and has defined a generation of artificial neural networks. The emergence of each new generation of artificial neural networks is facilitated by new challenges, such as the complexity of the tasks they are supposed to solve. The first generation of artificial neural networks includes the perceptron, developed in 1957 by Frank Rosenblatt. It is an early development of artificial intelligence and is considered the simplest type of artificial neural network. A significant limitation of this generation of neural networks is that they can only solve linearly separable problems. The second generation made it possible to solve the XOR task. The main difference between this generation of networks is their multilayer architecture, and the most famous example is the BackPropagation neural network proposed by Rumelhart, Hinton, and Williams in 1986 [1]. The next generation is characterized by an architecture with a large number of layers (Deep Neural Networks) [2]. It was the third generation that allowed for a significant breakthrough in many areas of artificial intelligence. It is worth noting that its emergence was also facilitated by the improvement of hardware and the development of data storage and processing technologies. This is very important, as this generation of networks requires significant computational costs and large samples of training data. The emergence of the fourth generation was facilitated by the complexity of artificial intelligence tasks. A feature of this generation of networks is

^{*}International Workshop on Computational Intelligence, co-located with the IV International Scientific Symposium "Intelligent Solutions" (IntSol-2025), May 01-05, 2025, Kyiv-Uzhhorod, Ukraine

[†]Corresponding author.

✉ yevgeniy.bodyanskiy@nure.ua (Y. Bodyanskiy); zaychenkoyuri@ukr.net (Y. Zaychenko); oleksii.kuzmenko@ukr.net (O. Kuzmenko); syncmaster@bigmir.net (H. Zaichenko)

ORCID 0000-0001-5418-2143 (Y. Bodyanskiy); 0000-0001-9662-3269 (Y. Zaychenko); 0000-0003-1581-6224 (O. Kuzmenko); 0000-0002-4630-5155 (H. Zaichenko)



© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

the ability to perform several different tasks using a single architecture. For this purpose, new architectures and methods have been created: large language models (LLMs), generative models, hybrid approaches to model building, etc.

Thus, the latest generation emphasizes the possibility of multitasking and versatility but also requires optimization of the complexity of the models, since the accuracy of the results depends on the computational costs.

2. Overview of problems in the application of deep learning neural networks

Deep learning neural networks play an important role in modern approaches to solving problems of prediction, classification, image processing, and natural language text analysis. Their universal approximation properties make them indispensable in many fields. However, the practical application of such powerful tools requires large amounts of training data, long time for model training, and significant computational costs. These are the key problems that can hinder the use of this type of neural network to solve complex problems and, consequently, obtain high accuracy results.

It is especially difficult to use deep neural networks when it is necessary to process non-stationary data streams transmitted in real time. In this case, radial basis function networks, Wang-Mendel, Takagi-Sugeno, or probabilistic neural networks can be an alternative. However, compared to deep neural networks, they have two significant drawbacks: lower accuracy of results and empirical selection of system parameters.

The above difficulties can be overcome by using an ensemble approach [3-8] to combine the work of models of different levels of complexity – from the first to the fourth generation. Ensembles allow the simultaneous use of different systems to solve a single problem in parallel in order to combine the output results. The use of the Bagging procedure [9] minimizes the error on the training set, even when the data is received online. The problem still remains the complexity of the system architecture, and thus the need for significant computational costs.

To simplify the ensemble model, we can apply Group Method of Data Handling (GMDH) [10, 11] to decompose the system and determine its optimal architecture. This method, called the “first deep learning method” by J. Schmidhuber [12], was used in the 1970s to create multilevel networks. It is worth noting that the advantage of using GMDH is not only to simplify the complexity of the system, but also to increase the accuracy of its results [13-17].

Therefore, we consider it expedient to investigate the proposed [18] Hybrid System of Computational Intelligence based on bagging and Group Method of Data Handling (HSCI-bagging). The aim of the investigation is to determine the optimal parameters of the hybrid system and the corresponding efficiency in the task of short- and middle-term stock index forecasting.

3. Overview of HSCI architecture based on Bagging and GMDH

In Fig. 1 the architecture of the proposed system is presented.

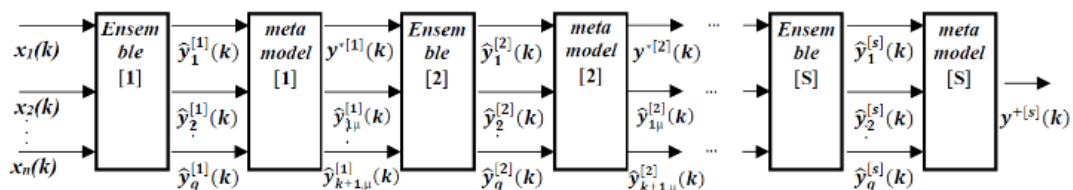


Figure 1: Architecture of HSCI based on bagging and GMDH

The architecture of the system contains 2S sequentially connected stacks, while odd stacks are formed by ensembles of parallel-connected subsystems that solve the same problem (recognition, prediction, etc.) and even ones are essentially learning metamodels that generalize the output signals of ensembles and form optimal results in the sense of the accepted criterion. The output signal of the first metamodel is the generalized optimal signal $y^{\zeta[1]}(k)$ and $(n-1)$ output signals $\hat{y}_{i\mu}^{[1]}(k), i=1,2,\dots,n-1(k)$ "best members of the ensemble". At their core, metamodels function as selection units in traditional GMDH systems, but not only select the best results from the previous stack, but also form the optimal solution based on these results.

Further, the output signals of the first metamodel are fed to the inputs of the second ensemble, which is completely similar to the first. The outputs of the second ensemble $\hat{y}_1^{[2]}(k), \hat{y}_2^{[2]}(k), \dots, \hat{y}_q^{[2]}(k)$ come to the second metamodel, which calculates the optimal signal $y^{\zeta[2]}(k)$ and $(n-1)$ $\hat{y}_{i\mu}^{[2]}(k)$ "closest" to it. The last S-th ensemble is similar to the first two, and the output of the last S-th metamodel is $y^{\zeta[S]}(k)$, which exactly corresponds to a priori established requirements for the quality of solving the problem under consideration.

Each of the ensembles contains q different computational intelligence systems that solve the same problem. There may still be simple neural networks such as a single-layer perceptron, radial-basis neural network (RBFN), counterpropagating neural network, etc., which do not use error backpropagation procedure for training, neuro-fuzzy systems such as ANFIS, Wang-Mendel or Takagi-Sugeno-Kang type, wavelet-neuro systems, neo-fuzzy neurons and others, the output signal of which linearly depends on the adapted parameters, which allows to use optimal speed learning algorithms.

4. HSCI-bagging learning algorithm

The input information, on the basis of which the system is configured, is a training selection of input signals:

$$\begin{aligned} & x(1), x(2), \dots, x(k), \dots, x(N); \\ & x(k) = (x_1(k), \dots, x_i(k), \dots, x_n(k))^T \in R^n \end{aligned} \quad (1)$$

and its corresponding scalar reference signals $y(1), \dots, y(k), \dots, y(N)$. On the basis of these observations, the elements of the first ensemble are tuned independently of each other, at the outputs of which q scalar signals $\hat{y}_p^{[2]}(k), p=1,2,\dots,q$, are formed, which are conveniently represented in the form of a vector $\hat{y}^{[1]}(k) = (\hat{y}_1^{[1]}(k), \dots, \hat{y}_p^{[1]}(k), \dots, \hat{y}_q^{[1]}(k))^T$. These signals are sent to the inputs of the first metamodel, at the outputs of which n sequences $\hat{y}^{\zeta[1]}(k), \hat{y}_{1\mu}^{[1]}(k), \dots, \hat{y}_{i\mu}^{[1]}(k), \dots, \hat{y}_{n-1,\mu}^{[1]}(k)$ the main of which is $y^{\zeta[1]}(k)$ while others are auxiliary. The main signal of the metamodel $y^{\zeta[1]}(k)$ is the union of the outputs of all members of the ensemble in the form of:

$$y^{\zeta[1]}(k) = \sum_{p=1}^q w_p^{\zeta[1]} \hat{y}_p^{[1]}(k) = \hat{y}^{[1]T}(k) w^{\zeta[1]} \quad (2)$$

where $w^{\zeta[1]} = (w_1^{\zeta[1]}, \dots, w_p^{\zeta[1]}, \dots, w_q^{\zeta[1]})^T$ – is a vector of adapted parameters-synaptic weights on which additionally restrictions are set on unbiasedness:

$$\sum_{p=1}^q w_p^{\zeta[1]} \zeta I_q^T w^{\zeta[1]} = 1, \quad (3)$$

where $I_q = (q \times 1)$ – is the vector of unities.

The problem of teaching the first metamodel is reduced to minimizing the standard quadratic criterion in the presence of additional constraints (3).

Thus, the problem of training the first metamodel can be solved using the standard method of penalty functions, which in this case reduces to minimizing the expression:

$$J(\mathbf{w}^{[1]}, \delta) = (Y(N) - \bar{Y}^{[1]}(N) \mathbf{w}^{[1]})^T (Y(N) - \bar{Y}^{[1]}(N) \mathbf{w}^{[1]}) + \delta^{-2} (I_q^T \mathbf{w}^{[1]} - 1) \quad (4)$$

where $Y(N) = (y(1), \dots, y(k), \dots, y(N))^T - (N \times 1)$ is a vector, $\bar{Y}^{[1]}(N) = (\hat{y}^{[1]}(1), \dots, \hat{y}^{[1]}(k), \dots, \hat{y}^{[1]}(N))^T - (N \times q)$ is a matrix, δ is the penalty coefficient.

Minimization (4) by $\mathbf{w}^{[1]}$ leads to the result:

$$\mathbf{w}^{[1]} = \lim_{\delta \rightarrow 0} \mathbf{w}^1(\delta) = \mathbf{w}^{LS[1]} + P^{[1]}(N) \frac{1 - I_q^T \mathbf{w}^{LS[1]}}{I_q^T P^{[1]}(N) I_q} I_q, \quad (5)$$

where $\mathbf{w}^{LS[1]}$ is a standard LSM estimate:

$$\mathbf{w}^{LS[1]} = (\bar{Y}^{[1]T}(N) \bar{Y}^{[1]}(N))^{-1} \bar{Y}^{[1]T}(N) Y(N) = P^{[1]}(N) \bar{Y}^{[1]T}(N) Y(N). \quad (6)$$

It was proved [4, 5] that the use of score (5) leads to results that are not inferior in accuracy to the best of the members of the first ensemble.

If observations from the training sample are processed sequentially online, it is advisable to use the least squares recurrent method in the form:

$$\left\{ \begin{array}{l} P^{[1]}(k+1) = P^{[1]}(k) - \frac{P^{[1]}(k) \hat{y}^{[1]}(k+1) \hat{y}^{[1]T}(k+1) P^{[1]}(k)}{1 + \hat{y}^{[1]T}(k+1) P^{[1]}(k) \hat{y}^{[1]}(k+1)}, \\ \mathbf{w}^{LS[1]}(k+1) = \mathbf{w}^{LS[1]}(k) + P^{[1]}(k+1) (y(k+1) - \hat{y}^{[1]T}(k+1) \mathbf{w}^{LS[1]}(k) \hat{y}^{[1]}(k+1)), \\ \mathbf{w}^{[1]}(k+1) = \mathbf{w}^{LS[1]}(k+1) + P^{[1]}(k+1) (I_q^T P^{[1]}(k+1) I_q)^{-1} \times (1 - I_q^T \mathbf{w}^{LS[1]}(k+1) I_q), \\ \mathbf{w}_p^{[1]}(0) = q^{-1} p = 1, 2, \dots, q. \end{array} \right. \quad (7)$$

or if a training sample is non-stationary we may use exponentially weighted recurrent LSM method:

$$\left\{ \begin{array}{l} P^{[1]}(k+1) = \frac{1}{\alpha} \left(P^{[1]}(k) - \frac{P^{[1]}(k) \hat{y}^{[1]}(k+1) \hat{y}^{[1]T}(k+1) P^{[1]}(k)}{\alpha + \hat{y}^{[1]T}(k+1) P^{[1]}(k) \hat{y}^{[1]}(k+1)} \right), \\ \mathbf{w}^{LS[1]}(k+1) = \mathbf{w}^{LS[1]}(k) + \frac{P^{[1]}(k) (y(k+1) - \hat{y}^{[1]T}(k+1) \mathbf{w}^{LS[1]}(k) \hat{y}^{[1]}(k+1))}{\alpha + \hat{y}^{[1]T}(k+1) P^{[1]}(k) \hat{y}^{[1]}(k+1)} \hat{y}^{[1]}(k+1), \\ \mathbf{w}^{[1]}(k+1) = \mathbf{w}^{LS[1]}(k+1) + P^{[1]}(k+1) (I_q^T P^{[1]}(k+1) I_q)^{-1} \times (1 - I_q^T \mathbf{w}^{LS[1]}(k+1) I_q), \\ \mathbf{w}_p^{[1]}(0) = q^{-1} p = 1, 2, \dots, q. \end{array} \right. \quad (8)$$

where $0 < \alpha \leq 1$ – forgetting factor.

To the parameters of the metamodel can be given meaning the levels of fuzzy membership to the optimal output signal by introducing additional restrictions on the non-negative values of these parameters, that is, in addition to the configurable parameters $\mathbf{w}^{[1]}$ we can also calculate the levels of this membership $\mu_p^{[1]} \geq 0, p = 1, 2, \dots, q$.

To do this, we introduce into consideration the extended Lagrange function:

$$L(\mu^{[1]}, \rho) = (Y(N) - \bar{Y}^{[1]}(N) \mu^{[1]})^T (Y(N) - \bar{Y}^{[1]}(N) \mu^{[1]}) + (I_q^T \mu^{[1]} - 1) - \rho^T \mu^{[1]}, \quad (9)$$

where $\rho - (q \times 1)$ – is vector of non-negative indefinite Lagrange multipliers.

Using the equations system by Kuhn-Tucker:

$$\begin{cases} \bar{V}_{\mu^{[1]}} L(\mu^{[1]}, \rho) = \vec{0}, \\ \partial L(\mu^{[1]}, \rho) / \partial = 0, \end{cases} \quad (10)$$

It's not difficult to get the solution in the form:

$$\begin{cases} \mu^{[1]} = p^{[1]}(N) (\hat{y}^T(N) Y(N) - 0,5 I_q + 0,5 \rho), \\ \frac{I_q^T p^{[1]}(N) \hat{y}^T(N) Y(N) - 1 + 0,5 I_q^T p^{[1]}(N) \rho}{0,5 I_q^T p^{[1]}(N) I_q}. \end{cases} \quad (11)$$

For finding vector of non-negative Lagrange multipliers, it's reasonable to apply Arrow-Hurwitz-Uzawa procedure:

$$(12)$$

where $P_{r+\hat{i}(\cdot)} \hat{i}$ is a projector to positive ortant, $\square_\rho(k)$ – learning rate, parameter.

First expression (12) after non-complex transformations may be presented in a more compact form:

$$\begin{aligned} \mu^{[1]}(k+1) = w^{\hat{i}^{[1]}}(k+1) - P^{[1]}(k+1) \frac{0,5 I_q^T P^{[1]}(k+1) \rho(k)}{I_q^T P^{[1]}(k+1) I_q} + 0,5 P^{[1]}(k+1) \rho(k) = \hat{i} \\ \hat{i} w^{\hat{i}^{[1]}}(k+1) + 0,5 \left(I_{qq} - \frac{P^{[1]}(k+1) I_q I_q^T}{I_q^T P^{[1]}(k+1) I_q} \right) \times P^{[1]}(k+1) \rho(k), \end{aligned} \quad (13)$$

where instead of least squares estimates $w^{LS^{[1]}}(k+1)$, the parameters of the metamodel $w^{\hat{i}^{[1]}}(k+1)$ are used, which simplifies the process of configuring it.

As a result of learning the first metamodel, the optimal signal $y^{\hat{i}^{[1]}}(k)$ is formed at its output, as well as q signals $\hat{y}_{p,\mu}^{[1]}(k)$ from which we choose $n-1$ (if $q \geq n$) with the highest levels of fuzzy membership $\mu_p^{[1]}$, which subsequently in the form of $(n \times 1)$ – vector are fed to the input of the second ensemble, the outputs of which go to the inputs of the second metamodel, and so on. The process of increasing the number of ensembles and metamodels continues until the required accuracy of the last metamodel with the output $y^{\hat{i}^{[s]}}(k)$ is achieved, or the value of the criterion minimized for the bagging model begins to increase, i.e. $\bar{\epsilon}^2(y^{\hat{i}^{[s+1]}}(k)) \geq \bar{\epsilon}^2(y^{\hat{i}^{[s]}}(k))$.

5. Experimental investigations

The dependence of the prediction accuracy of the hybrid HSCI-bagging system on the parameters in the prediction task was investigated. The obtained forecasting results were compared with the results of BackPropagation and GMDH by MSE and MAPE criteria.

For the experiments, the dataset was divided into training, validation, and test subsamples. The test subsample was used to evaluate the accuracy of the forecast and was always constant - the last 50 points of the dataset. The ratio of training and test data sets varied across the experiments and had the

following values (in percentage): 60/40, 70/30, 80/20. The forecasting intervals also changed: short-term (3, 5, 7 days) and middle-term (10, 20, 30 days).

The historical data of the Close indicator of the DAX P index (^GDAXI) [19] for the period 24.03.2024-20.03.2025 were used as data set. The dynamics of changes of the Close indicator is shown in Fig. 2.

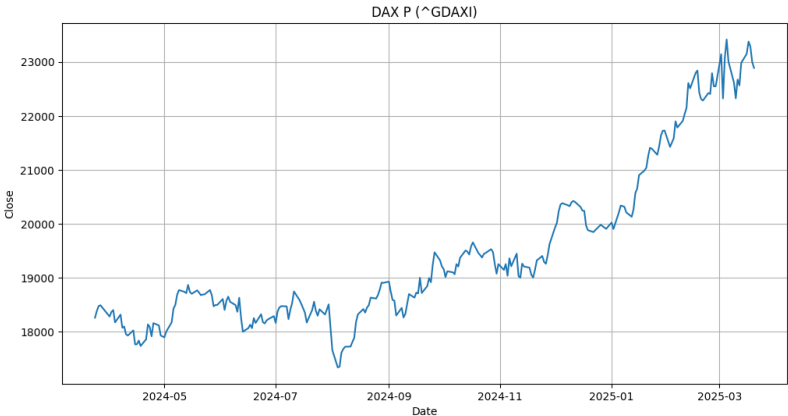


Figure 2: Dynamics of the Close indicator of the DAX P stock index

The correlation coefficients for the values of the Close indicator were also calculated, and correlogram was built (Fig. 2).

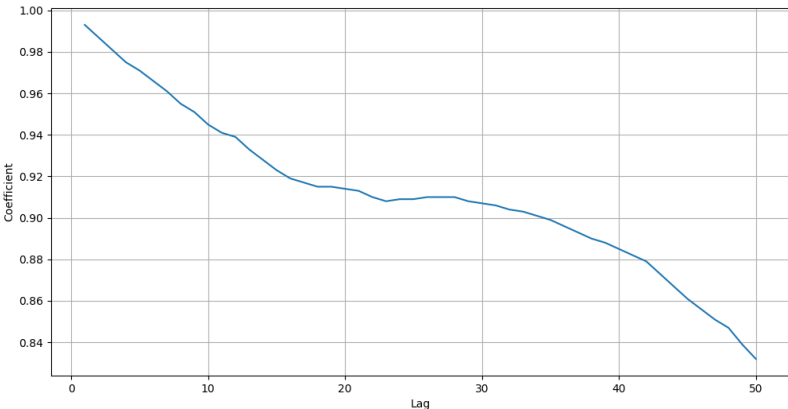


Figure 3: Correlogram of the Close indicator of the DAX P stock index

Analysing the correlogram, the data is highly correlated, with a correlation coefficient of at least 0.8 for the 50th lag.

The first series of experiments was conducted on short-term intervals for the BackPropagation, GMDH, and HSCI-bagging models. For each of the models, the ratio train/test was changed at each interval, and the optimal parameters (number of inputs, number of layers and number of neurons on each layer) for the BackPropagation network were determined. The metrics of the best results are shown in Tables 1-3.

Table 1
Values of forecast quality metrics with 3-day interval

Metrics	BackPropagation	GMDH	HSCI-bagging
MSE	979294.79	895189.3	306563.12
MAPE	3.82	3.14	1.89

Table 2
Values of forecast quality metrics with 5-day interval

Metrics	BackPropagation	GMDH	HSCI-bagging
MSE	2087809.67	1800193.41	1565304.41
MAPE	5.11	5.02	4.67

Table 3
Values of forecast quality metrics with 7-day interval

Metrics	BackPropagation	GMDH	HSCI-bagging
MSE	2513701.86	3349316.64	1976780.36
MAPE	5.91	6.38	5.51

Charts of short-term forecasts for HSCI-bagging models are shown in Fig. 4-6.

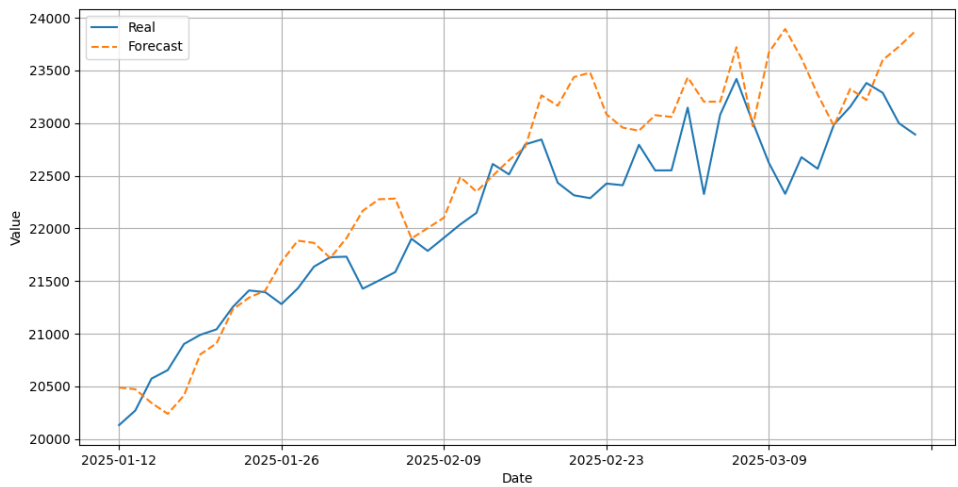


Figure 4: Comparative chart of real values and forecast of HSCI-bagging with an interval of 3 days

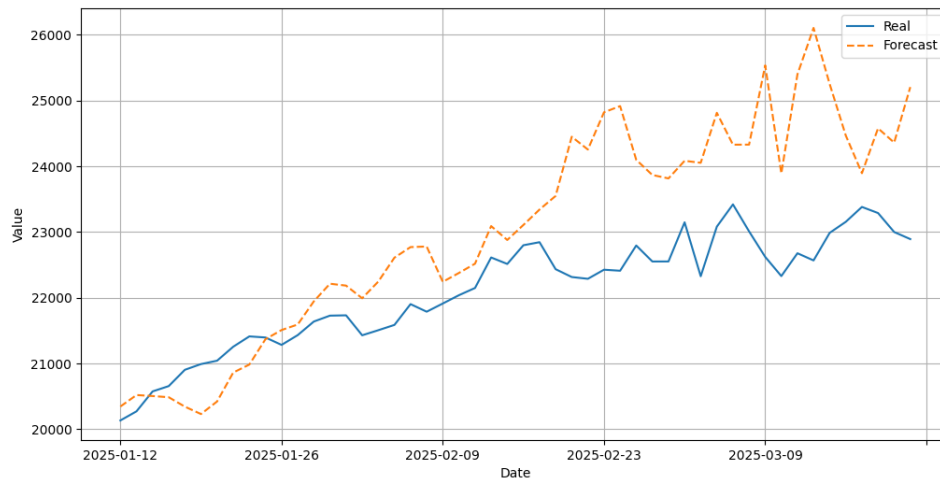


Figure 5: Comparative chart of real values and forecast of HSCI-bagging with an interval of 5 days

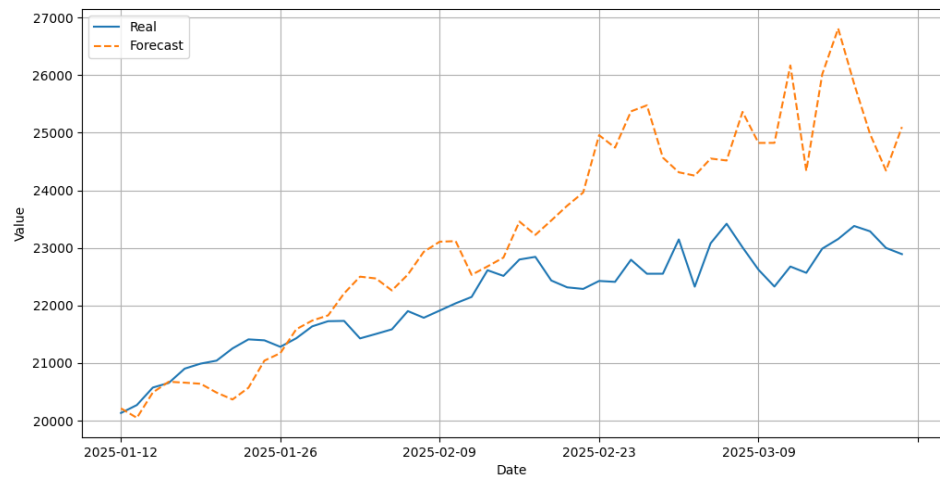


Figure 6: Comparative chart of real values and forecast of HSCI-bagging with an interval of 7 days

For the analysis of the forecasting and model evaluation results, comparative bar charts of MSE (Fig. 7) and MAPE (Fig. 8) were built.

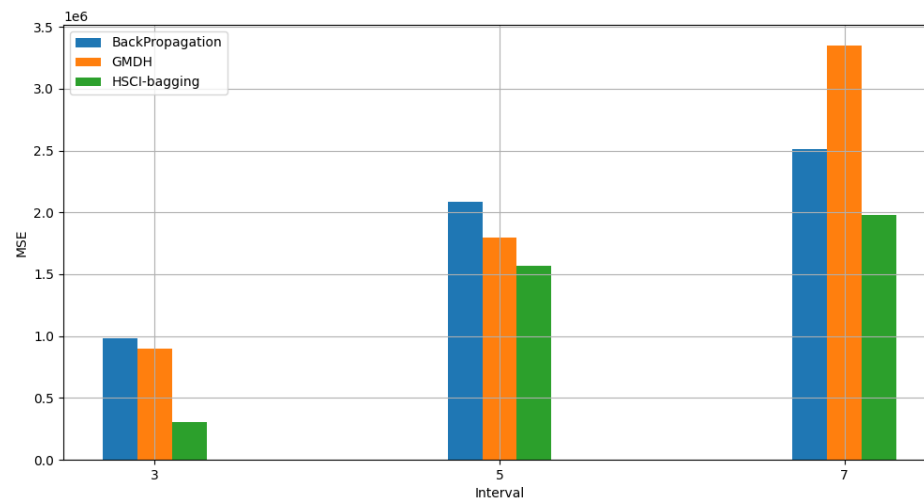


Figure 7: Comparative bar charts of MSE values of all models for short-term forecasting

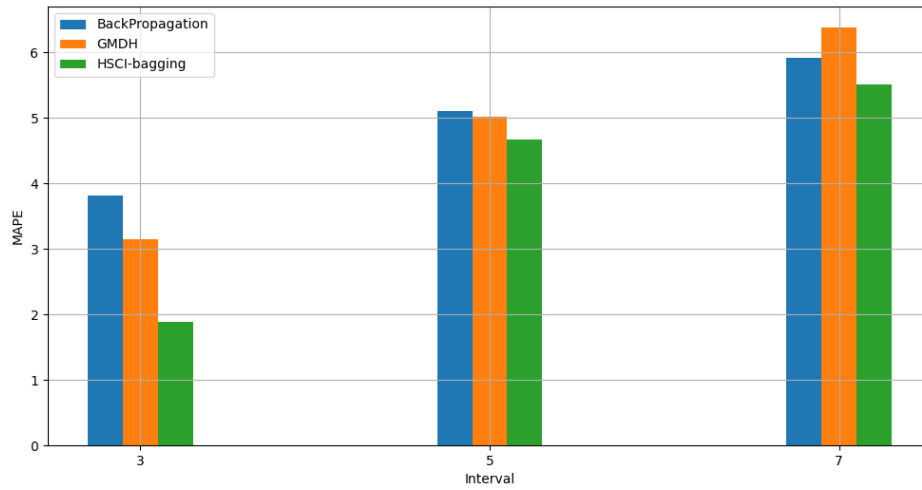


Figure 8: Comparative bar charts of MAPE values of all models for short-term forecasting

The next series of experiments was conducted at middle-term intervals. The quality criteria for the obtained forecasts are shown in Tables 4-6 and charts of middle-term forecasts for HSCI-bagging models are shown in Fig. 9-11.

Table 4

Values of forecast quality metrics with 10-day interval

Metrics	BackPropagation	GMDH	HSCI-bagging
MSE	4443943.98	4244875.38	2943590.87
MAPE	7.38	7.22	7.04

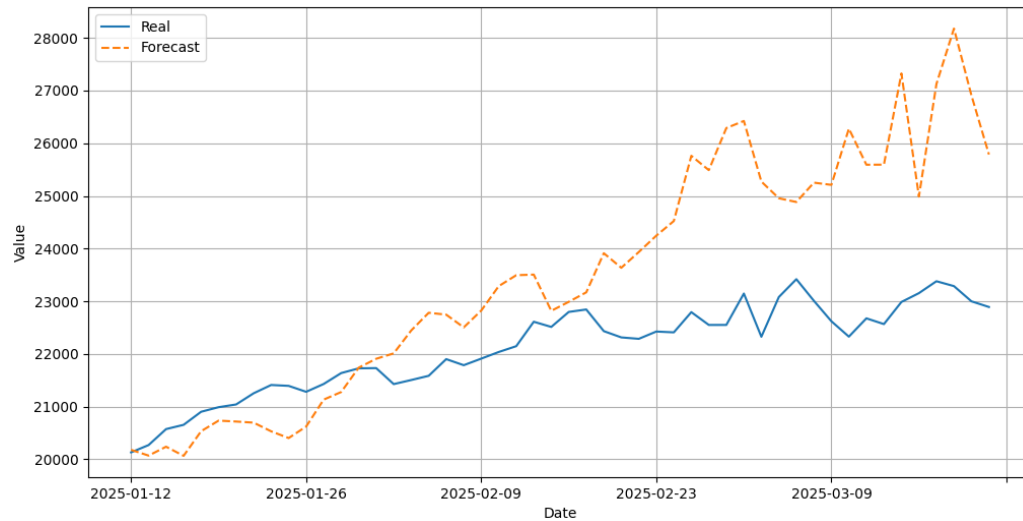


Figure 9: Comparative chart of real values and forecast of HSCI-bagging with an interval of 10 days

Table 5

Values of forecast quality metrics with 20-day interval

Metrics	BackPropagation	GMDH	HSCI-bagging
MSE	4631698.32	3853634.31	3821746.29

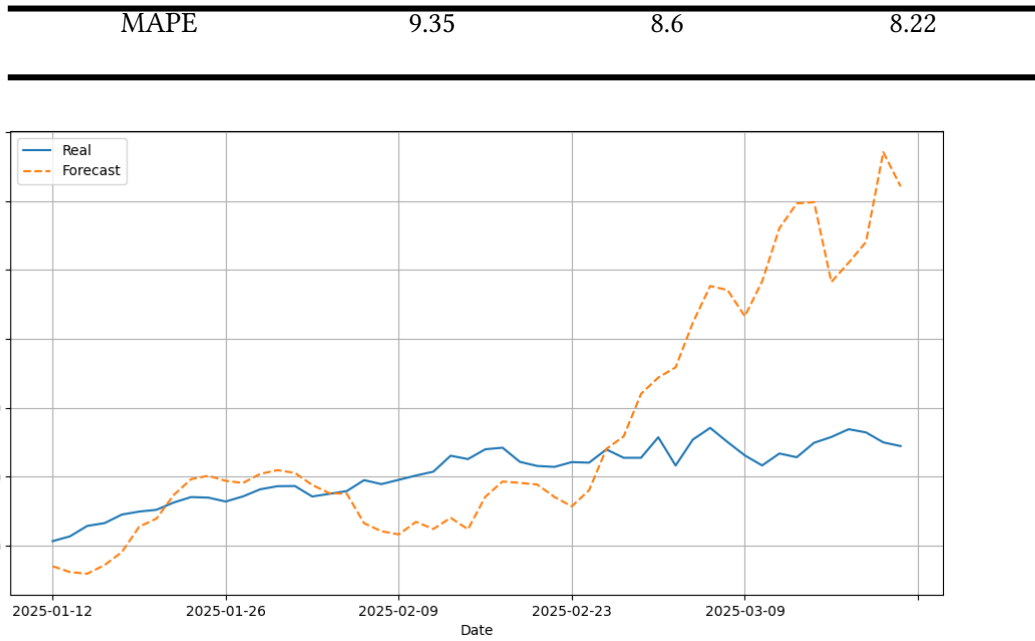


Figure 10: Comparative chart of real values and forecast of HSCI-bagging with an interval of 20 days

Table 6

Values of forecast quality metrics with 30-day interval

Metrics	BackPropagation	GMDH	HSCI-bagging
MSE	9164351.68	7973722.81	7291331.07
MAPE	11.64	11.14	11.02

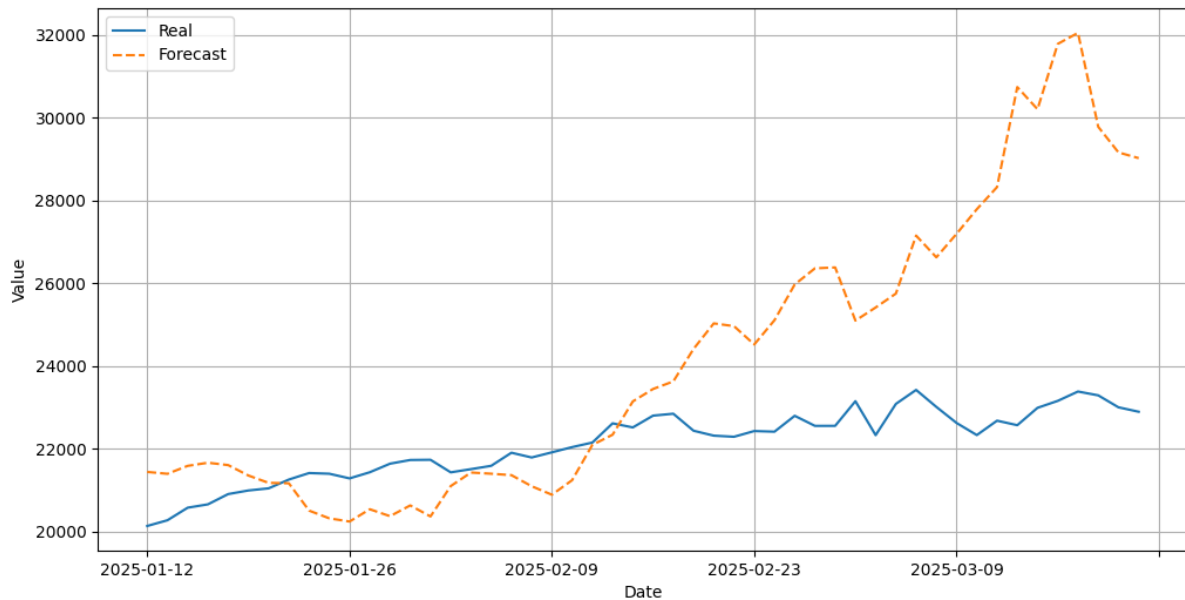


Figure 11: Comparative chart of real values and forecast of HSCI-bagging with an interval of 30 days

For the analysis of the forecasting results and model evaluation, comparative bar charts were constructed for two key error metrics: MSE and MAPE. The bar chart representing MSE is shown in Fig. 12, and the bar chart illustrating MAPE is shown in Fig. 13. Presenting the comparison of metrics in this way clearly demonstrates how different the predictions of different models are.

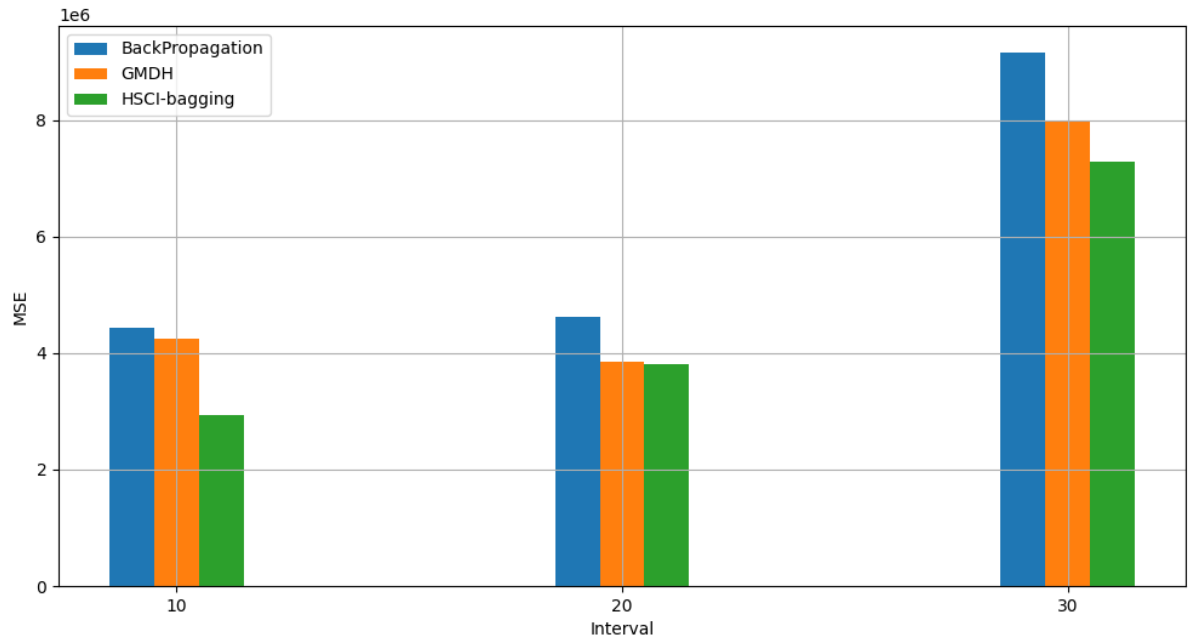


Figure 12: Comparative bar charts of MSE values of all models for middle-term forecasting

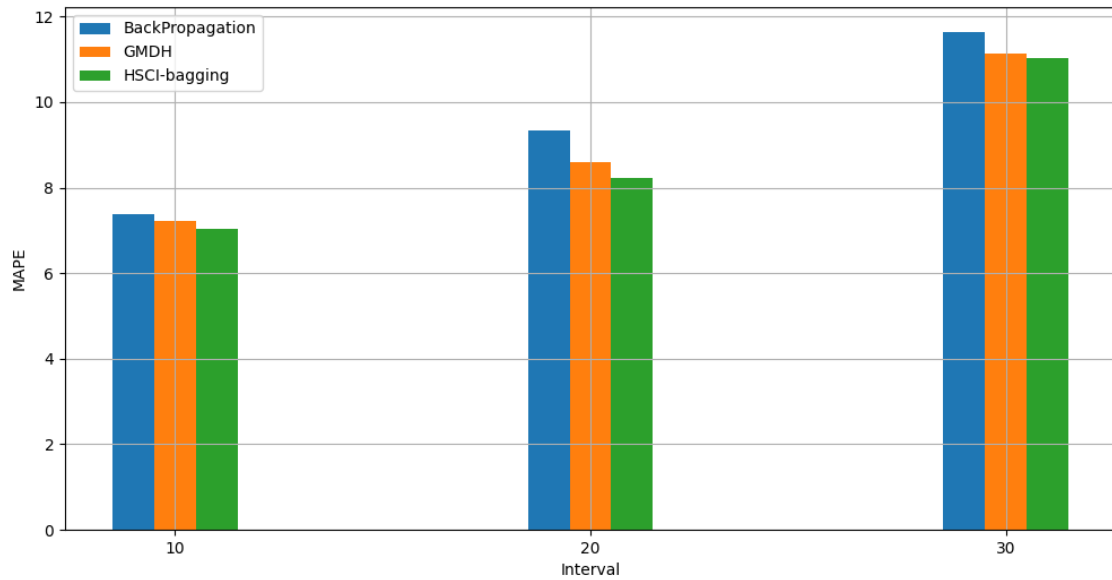


Figure 13: Comparative bar charts of MAPE values of all models for middle-term forecasting

At the final stage of the experiments, Table 7 was built to analyze the HSCI-bagging models.

Table 7

Summary table of experimental results for HSCI-bagging

Interval	Ratio train/test (%)	MSE	MSPE
3	60/40	306563.12	1.89
5	60/40	1565304.41	4.67
7	60/40	1976780.36	5.51
10	60/40	2943590.87	7.04

20	70/30	3821746.29	8.22
30	70/30	7291331.07	11.02

To visually represent the dependence of the MSE and MAPE criteria values on the forecasting interval, the respective charts were constructed (Fig. 14 and Fig. 15).

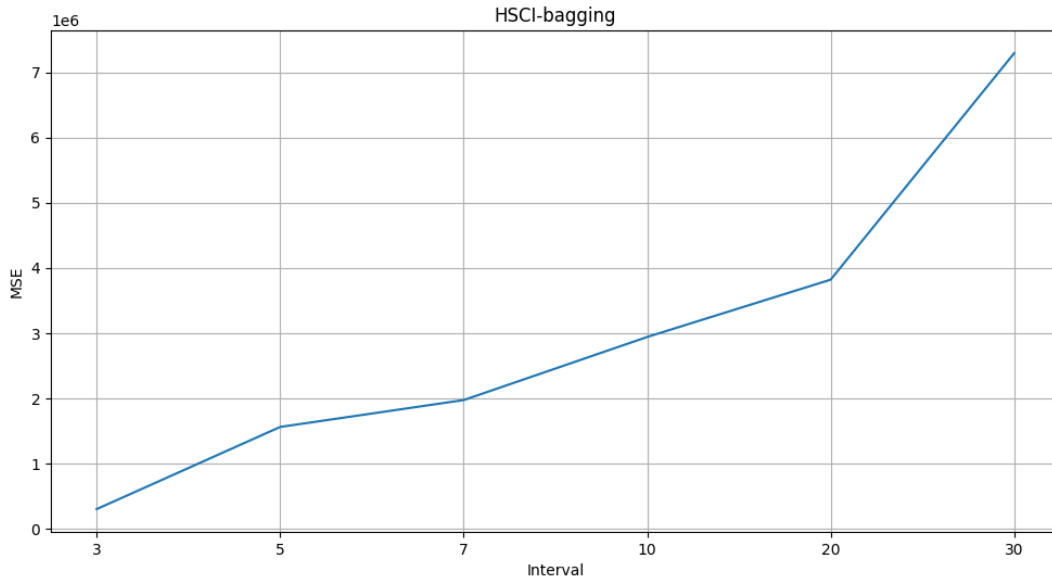


Figure 14: Chart of dependence of MSE values on the forecasting interval for HSCI-bagging

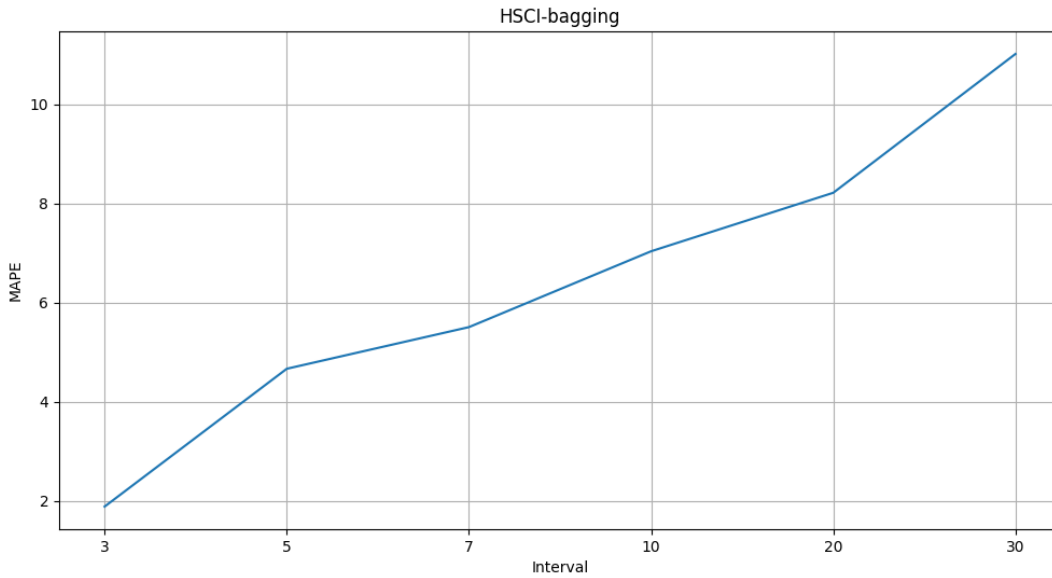


Figure 15: Chart of dependence of MAPE values on the forecasting interval for HSCI-bagging

Based on the analysis of the experimental results, it can be said that the hybrid HSCI-bagging system is quite effective compared to BackPropagation and GMDH. Table 7 shows that the network requires a larger amount of training data to improve the forecast accuracy at middle-term intervals. Looking at Figs. 14 and 15, it can be said that the forecast accuracy decreases rapidly when

forecasting for an interval of 20 days or more. It is also worth noting that as the forecasting interval increases, the accuracy of the MAPE criterion decreases more slowly than the MSE criterion.

6. Conclusion

This article investigates the problem of short- and medium-term forecasting of stock indices, in particular the Close indicator of the DAX P index, using intelligent methods of different generations. A comparative experimental study of three models - BackPropagation, GMDH and HSCI-bagging - has led to a number of important conclusions:

- The hybrid computational intelligence system HSCI-bagging demonstrated the best forecasting results for all criteria (MSE, MAPE) at different intervals, which indicates its high generalization ability and resistance to changes in data.
- The BackPropagation neural network showed the worst results, which may be due to high sensitivity to training parameters, lack of data, or local minima in the optimization process.
- The dependence of the forecasting quality on the size of the training sample was established: as the forecast horizon increases, the need for more historical data increases to ensure stable accuracy.
- Hybrid approaches based on bagging demonstrate high adaptability and potential for integration with other intelligent methods, including ensemble structures.

Based on the results, the following recommendations for further development can be made:

- Integration of time series processing methods (e.g., wavelet transform, EDA, STL decomposition) with hybrid intelligent systems to improve data preprocessing and highlight hidden patterns.
- Use of deep ensemble models based on LSTM, GRU, or Transformer architectures, which can be combined with GMDH or other evolutionary approaches within meta-models.
- Adaptive real-time optimization of HSCI system parameters using reinforcement learning or evolutionary strategies.
- Development of multimodal models that combine numerical, textual, and graphical data (e.g., news, tweets, corporate reports) to build more contextually aware forecasts.
- Incorporating risk-oriented metrics such as Value-at-Risk (VaR) or Conditional Value-at-Risk (CVaR) into the process of evaluating forecasting performance to increase the practical value of decisions in financial applications.

Thus, the further development of the forecasting financial indices lies in the development of adaptive, hybrid and multilevel models that combine high forecasting accuracy with scalability and practical applicability to real market conditions.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

- [1] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by backpropagating errors, *Nature* 323 (1986) 533–536. doi:10.1038/323533a0.
- [2] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016. URL: <http://www.deeplearningbook.org>.
- [3] L. K. Hansen, P. Salamon, Neural network ensembles, *IEEE Trans. Pattern Anal. Mach. Intell.* 12 (1990) 993–1000.

- [4] Ye. Bodyanskiy, I. Pliss, Adaptive generalized forecasting of multivariate stochastic signals, in: Proc. Latvian Sign. Proc. Int. Conf., vol. 2, Riga, 1990, pp. 80–83.
- [5] Ye. V. Bodyanskiy, I. A. Rudneva, On one adaptive algorithm for detecting discords in random sequences, *Autom. Remote Control* 56 (1995) 1439–1443.
- [6] A. J. C. Sharkey, On combining artificial neural nets, *Connect. Sci.* 8 (1996) 299–313.
- [7] S. Hashem, Optimal linear combination of neural networks, *Neural Networks* 10 (1997) 599–614.
- [8] U. Naftaly, N. Intrator, D. Horn, Optimal ensemble averaging of neural networks, *Network: Comput. Neural Syst.* 8 (1997) 283–296.
- [9] L. Breiman, Bagging Predictors, Techn. Report №421, Dept. of Statistics, Univ. of California, Berkeley, CA, 1994, 19 p.
- [10] A. G. Ivakhnenko, V. G. Lapa, *Cybernetic Forecasting Devices*, Naukova Dumka, Kyiv, 1965, 216 p. (in Ukrainian).
- [11] A. G. Ivakhnenko, G. A. Ivakhnenko, J. A. Mueller, Self-organization of the neural networks with active neurons, *Pattern Recogn. Image Anal.* 4 (1994) 177–188.
- [12] J. Schmidhuber, Deep learning in neural networks: An overview, *Neural Networks* 61 (2015) 85–117.
- [13] Y. Zaychenko, Ye. Bodyanskiy, O. Tyshchenko, O. Boiko, G. Hamidov, Hybrid GMDH-neuro-fuzzy system and its training scheme, *Int. J. Inf. Theories Appl.* 24 (2018) 156–172.
- [14] Y. Zaychenko, Ye. Bodyanskiy, O. Boiko, G. Hamidov, Evolving Hybrid GMDH NeuroFuzzy Network and Its Application, in: Proc. Int. Conf. IEEE-SAIC 2018, Kyiv, Ukraine, IASA, 8–11 Oct. 2018.
- [15] Ye. Bodyanskiy, N. Kulishova, Y. Zaychenko, G. Hamidov, Spline-Orthogonal Extended Neo-Fuzzy Neuron, in: Proc. Int. Conf. CISP–BMEI 2019.
- [16] Ye. Bodyanskiy, Y. Zaychenko, O. Boiko, G. Hamidov, A. Zelikman, The Hybrid GMDH-Neo-fuzzy Neural Network in Forecasting Problems in Financial Sphere, in: *Advances in Intelligent Computing*, Springer, 2020, vol. 1075, pp. 221–225.
- [17] Ye. Bodyanskiy, O. Vynokurova, I. Pliss, Hybrid GMDH-neural network of computational intelligence, in: Proc. 3rd Int. Workshop on Inductive Modeling, Krynica, Poland, 2009, pp. 100–107.
- [18] Y. Bodyanskiy, O. Kuzmenko, H. Zaichenko, Y. Zaychenko, Application of Hybrid Neural Networks based on bagging and Group Method of Data Handling for forecasting, in: Proc. 2023 IEEE 18th Int. Conf. on Computer Science and Information Technologies (CSIT), Lviv, Ukraine, 2023, pp. 1–6. doi:10.1109/CSIT61576.2023.10324161.
- [19] Yahoo Finance, DAX P (^GDAXI) Stock Price, News, Quote and History, 21 Mar. 2025. URL: <https://finance.yahoo.com/quote/%5EGDAXI/>. Accessed: 21 Mar. 2025.