UDC 681.327.12 (045)
DOI:10.18372/1990-5548.54.12323

[1]O. I. Chumachenko,
[2]A. T. Kot

STRUCTURAL-PARAMETRIC SYNTHESIS OF HYBRID NEURAL NETWORKS ENSEMBLES

[1,2]National Technical University of Ukraine "Ihor Sikorsky Kyiv Polytechnic Institute," Kyiv, Ukraine
E-mails: [1]chumachenko@tk.kpi.ua, [2]anatoly.kot@gmail.com

***Abstract*—*It is considered the approach to the design of the ensemble of neural networks, where a collection of a finite number of neural networks is trained for the same task, then their results of the given task solution are combined. It is proposed an algorithm of optimal choice of neural networks topologies and their quantity for their inclusion as a member in ensemble. The further refinement of ensemble composition is done with help pruning operation. The output of an ensemble is a weighted average of the outputs of each network, with the ensemble weights determined as a function of the relative error of each network determined in training. It is presented a novel approach to determine the ensemble weights dynamically as part of the training algorithm. The weights are proportional to the certainty of the respective outputs.*

Index Terms—Neural networks; ensemble; training; optimization; topology.

I. INTRODUCTION

An ensemble of neural networks (NN) is called a group of topologies, united into a single structure, which may differ in architecture, learning algorithm, training criteria, and types of generating neurons [1]. In another variant, the term ensemble means "united model", the output of which is a functional combination of individual models outputs [1]. The construction of classifier ensembles is an active field of research in machine learning because of the improvements in classification accuracy that can be obtained by combining the decisions made by the units in the ensemble.

Input data can be broken down into certain groups for processing by different NNs or applied to all networks at the same time.

Forming of NN ensembles requires a solution of two stages problem – the qualitative learning of each NN which is supposed to include to the ensemble and their optimal association. The known algorithms are divided into two classes: algorithms that for new classifiers change the distribution of learning examples based on the accuracy of the previous models (boosting), and those in which new members of the ensemble learn independently of others (bagging).

II. PROBLEM REVIEW

The main algorithms of the NN ensemble association and their disadvantages are shown in the Table I [1].

In contrast to [2], in this work, instead of separate NNs, modules of neural networks are used.

The necessity of the modulus principle applying in the hybrid NN of ensemble structure is determined by the following:

– heterogeneity of the data of the training sample, which leads to the inability of a one-module NN to correctly approximate the necessary dependence;

– the complexity of the algorithm of the solvable problem, which requires a multimodal structure;

– characteristics difference of the errors function on the various fragments of the training sample;

– need to accumulate the knowledge of experts in NN modules learning.

For new, there are serial and parallel types of structures for the construction of NN ensembles. Investigations of the serial structure of the of modules connection into the ensemble are currently absent. In work [3] the examples of serial construction of modules structure networks are presented. An example of a parallel modules structure is presented in work [4].

The main difficulty of networks ensemble association is the training of all components for the problem solution. In order to increase the effectiveness of learning, each NN is learnt separately (if possible), and then united into a single structure. However, in the case where the training algorithms of selected topologies belong to different classes, the synchronous training of all modules included in the ensemble is required, and therefore it is necessary to develop a single algorithm for all ensembles modules adjustment.

Let's consider the principles of hybrid NN ensembles constructing of seriel and parallel structures.

III. MODULE CONNECTION TYPES

The sireal ensemble organization structure consists in supplying the output data of one module to the inputs of another module. A similar structure is used to restore the input data or to improve their differences (normalization) for execution of the main task (approximation, classification, etc.).

The general scheme of modules serial connection of the is shown in Fig. 1.

TABLE I

CHARACTERISTICS OF ALGORITHMS

Technology	**Methodology for obtaining the result**	**Disadvantages**
Static structures		
Averaging over ensemble	Linear combination of NN output signals	1. Dependence of the result on the correct determination of the competence of the NN. 2. Increasing the complexity of the algorithm by applying algorithms of "data noise emissions" correction.
Boosting	Each new NN is based on the results of previously built neuron networks	1. The presence of more examples of learning sample. 2. The degeneration of the NN ensemble into a complex inefficient neural network structure that requires a large amount of computing resources. 3.The last NNs learn on the "most complex" examples.
Stacking	Applying the concept of meta learning	1. The complexity of the theoretical analysis through a set of sequentially shaped models. 2. Possible growth of meta model levels, which can lead to a rapid depletion of computing resources.
Bagging	Formation of the NNs set on the basis of the set of subsets of the learning sample and the subsequent combining of the results of the SNN work	1. Additional computational expenses associated with the need to form a large number of learning sample subsets. 2. The subsets of the examples differ from each other, but are not independent, since they all are based on the same set. 3. The algorithm requires a large amount of data for adjustment and learning.
Dynamic structures		
Mixing of expert opinions	Integration of expert knowledge through the use of the gateway network	1. The algorithm demands the computing resources in the breakdown of the output area. It is possible to create a large number of areas, which will lead to excessive clustering of area and will create a larger group of basic NNs with a complex mechanism of interaction with the networks of the gateway. 2. Learning and adjusting the hierarchical model represents a complex computational process. The learning process based on the stochastic gradient is based on the adjustment of the weight coefficients of the NNs, the network of the gateway of the first and second levels, which leads to a complicated algorithm of complex optimization of the total NN machine.

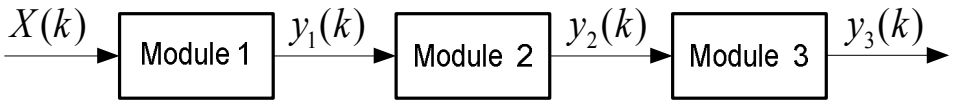

Fig. 1. Serial connection of modules

An ensemble in which the input data is applied simultaneously to the inputs of all modules that form the hybrid NN is called parallel. The main element of such structure is the "layer of association", which is responsible for aggregating the results of the various ensemble components. The general structure of the parallel modules ensemble of NN is shown in Fig. 2.

The main disadvantage of using a parallel ensemble is an overly complex learning algorithm with probabilistic convergence.

Based on the analysis conducted in this paper, it is proposed to synthesize the hybrid topology in the form of a series-parallel ensemble of NN modules, which is the most generalizing structure. The selected conceptual model is shown in Fig. 3.

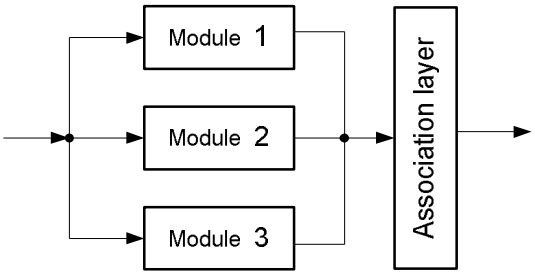

Fig. 2. Parallel connection of modules

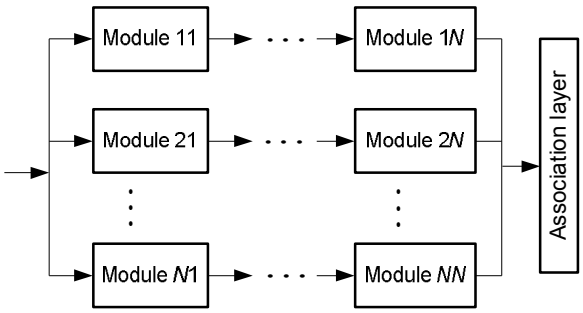

Fig. 3. Serial-parallel structure of ensemble of NN modules

IV. PROBLEM STATEMENT

"A necessary and sufficient condition for an ensemble of classifiers to be more accurate than any of its individual members is if the classifiers are accurate and diverse." Since the diversity of the ensemble decreases with the increase in accuracies of ensemble members, the key to the success of any ensemble learning method is the appropriate handling of the trade-off between accuracy and diversity.

Many approaches have been proposed to create accurate and diverse ensembles. Examples include bagging [5], boosting [8], random forests [6], the random subspace method [9] and random decision trees [7]. In most ensemble methods, the diversity and accuracy are acquired by manipulating subsets of data points or features. One problem with these ensembling approaches is that they tend to construct unnecessarily large ensembles, which requires a large amount of memory to store the trained classifiers and decreases the response time for prediction. Ensemble pruning, or selective ensembles, is a technique that tackles this problem by choosing a subset of individual classifiers from a trained ensemble to form a subensemble for prediction. The classifiers in the subensemble need to be carefully chosen so that it is small enough to reduce the memory requirement and response time with predictive accuracy that is similar to or better than the original ensemble.

Based on analysis of Table I it can by shown that bagging has advantages before others:

– Bagging reduces variance or model nconsistency over diverse data sets from a given distribution, ithout increasing bias, which results in a reduced error and enhanced stability.

– The other benefit of using bagging is related to the model selection. Since bagging transforms a group of over-fitted neural networks into a better-than perfectly-fitted network, the tedious time consuming model selection is no longer required. This could even offset the computational overhead needed in bagging that involves training many neural networks.

– Bagging is very robust to noise.

– Parallel execution: although the boosting algorithm has better generalization ability than the bagging algorithm, the bagging algorithm has the benefit of training ensembles independently, hence in parallel.

Let $D = \{d_1, ..., d_N\}$ be a set of N data points where $d_i = \{(x_i, y_i) \mid \text{and} \in [1, N]\}$ is a pair of input features and label that represents the ith data point, $C = \{c_1, ..., c_M\}$ be a set of M classifiers where $c_i(x_j)$ gives the prediction of the ith classifier on the jth data point, $V = \{v^{(1)}, ..., v^{(N)} | v^{(i)} = [v_1^{(i)}, ..., v_L^{(i)}]$, and $\in [1, N]\}$ be a set of vectors where $v_j(i)$ is the number of predictions for the jth label of the ith data point of an ensemble combined with majority voting, and L is the number of output labels.

It is necessary, based on the accuracy and variety of classifiers $C = \{c_1, ..., c_M\}$, to select members to form an ensemble, having a test set of data and assuming that the networks are previously trained on bootstrap samples.

V. PROBLEM SOLUTION

For the begining it is necessary to select the candidates from the set of NN for their inclusion into ansemble.

From our point of view it must be hybrid NN, for example modules of NNs [3], [4], whire is presented their structure.

Equal models become diverse when learning from different data sets. If we have at our disposal only one set of m examples, then the different subsamples with close statistics can be obtained by applying a bootstrap [17] – a random sample with a return.

The main idea of the bootstrap is to repeatedly retrieve repeated samples from the empirical distribution by the method of Monte Carlo statistical tests. Namely, we take a finite set of n terms of the original sample $x_1, x_2, ..., x_{n-1}, x_n$, from which it is "stretched" at each step of n consecutive iterations using a random number generator uniformly distributed on the interval $[1, n]$ an arbitrary element x_k, which again "returns" to the original sample (that is, it can be retrieved).

So preliminary stage of ensemble building is the creation of basic classifiers which must be independet.

These classifiers are learnt on independent data sets. As result we have the following algorithm:

1) A set of training examples is given (x_1, y_1), …, (x_m, y_m) tagged $y \in \{1, ..., k\}$.

2) Get T bootstrap sampling D_t.

3) Independently (in parallel) to train t classifiers h_t, everyone in their sample D_t.

In bagging, only a subset of examples typically appears in the bag which will be used in training the classifier. Out-of-bag error provides an estimate of the true error by testing on those examples which did not appear in the training set. Formally, given a set T of examples used in training the ensemble, let t be a set of size $|T|$ created by a random sampling of T with replacement, more generally known as a bag. Let s be a set consisting of $T - (T \cap t)$. Since s consists of all those examples not appearing within the bag, it is called the out-of-bag set. A classifier is trained on set t and tested on set s. In calculating the voted error of the ensemble, each example in the training set is classified and voted on by only those classifiers which did not include the example in the bag on which that classifier was trained. Because the out-of-bag elements, by definition, were not used in the training set, they can be used to provide an estimate of the true error.

We have developed an algorithm which appears to provide a reasonable solution to the problem of deciding when enough classifiers have been created for an ensemble. It works by first smoothing the out-of-bag error graph with a sliding window in order to reduce the variance [19].

Only a fraction of the trees in the ensemble are eligible to vote on any given item of training data by its being "out-of-bag" relative to them. Consider the meaning of diversity.

Given two classifiers c_i and c_j, where $N^{(01)}$ denotes the number of data points incorrectly predicted by c_i but correctly predicted by c_j, and $N^{(10)}$ is the opposite of $N^{(01)}$, the diversity of ci and c_j, denoted by $\mathrm{Div}_{i,j}$, is the ratio between the sum of the number of data points correctly predicted by one of the classifiers only and the total number of data points, as given in equation

$$\mathrm{Div}_{i,j} = \frac{N^{(01)} + N^{(10)}}{N}. \quad (1)$$

A classifier c_i's diversity contribution to an ensemble, denoted by ConDiv_i, is the sum of the diversities between c_i and each other classifier in the ensemble (excluding ci because according to equation (1) a classifier's diversity to itself is zero), as given in equation

$$\mathrm{ConDiv}_i = \sum_{j=1}^{M} \mathrm{Div}_{i,j}. \quad (2)$$

For the two-class classification problem, the variety contribution of classifier c_i

$$\mathrm{ConDiv}_i = \frac{1}{N}\sum_{k=1}^{N}\left(M - v^{(i)}_{c_i(x_k)}\right), \quad (3)$$

where N is the number of data points; M is the total number of classifiers; $v^{(i)}_{c_i(x_k)}$ is the number of classifiers that agree with c_i in prediction (including itself); $\left(M - v^{(i)}_{c_i(x_k)}\right)$ is the number of classifiers that disagree with c_i in prediction. In a two-class learning task, each disagreement is counted once when calculating the diversity contribution of c_i, as defined in equation (2). Thus the sum of the disagreements on all data points divided by N is exactly equal to equation (2).

In general an individual classifier's prediction on the data points can be divided into four exclusive subsets:

1) the subset in which the individual classifier predicts correctly and is in the minority group (ensemble predicts incorrectly);

2) the subset in which the individual classifier predicts correctly and is in the majority group (ensemble predicts correctly);

3) the subset in which the individual classifier predicts incorrectly and is in the minority group (ensemble predicts correctly);

4) the subset in which the individual classifier predicts incorrectly and is in the majority group (ensemble predicts incorrectly).

In work [20] for designing a heuristic metric for evaluating individual contributions of ensemble members it is determined the rules of for evaluating contributions of predictions: 1) correct predictions make positive contributions (correct predictions that are in the minority group make more positive contributions than correct predictions that are in the majority group); 2) incorrect predictions make negative contributions (incorrect predictions that are in the minority group make less negative contributions than incorrect predictions that are in the majority group).

The individual contribution of a classifier c_i is therefore defined as:

$$\mathrm{IC}_i = \sum_{j=1}^{N} \mathrm{IC}_i^{(j)}, \quad (4)$$

where $\mathrm{IC}_i^{(j)}$ is c_i's contribution on the jth data point d_j.

When $c_i(x_j)$ equals y_j, which means c_i makes correct predictions on d_j, if $c_i(x_j)$ is in the minority group (the first subset), $\mathrm{IC}_i^{(j)}$ is defined as:

$$\mathrm{IC}_i^{(j)} = 2v_{\max}^{(j)} - v_{c_i(x_j)}^{(j)}, \quad (5)$$

where $v_{\max}^{(j)}$ is the number of majority votes on d_j and $v_{c_i(x_j)}^{(j)}$ is the number of predictions on $c_i(x_j)$, as defined before.

When $c_i(x_j)$ equals y_j and $c_i(x_j)$ is in the majority group (in this case $v_{c_i(x_j)}^{(j)} = v_{\max}^{(j)}$) (the second subset), $\mathrm{IC}_{(j)}^{i}$ is defined as:

$$\mathrm{IC}_{(j)}^{i} = v_{\mathrm{sec}}^{(j)}, \quad (6)$$

where $v_{\mathrm{sec}}^{(j)}$ is the second largest number of votes on the labels of d_j. $\left(v_{\mathrm{sec}}^{(j)} - v_{\max}^{(j)}\right)$ is an estimation of the "degree of positive contribution" in this case.

It is clear, if the majority of classifiers predicts correctly with a classifier on d_j, this classifier's contribution is not very valuable because without its prediction, the ensemble would still be correct on d_j (assuming no tie). Note that $\left(v_{\mathrm{sec}}^{(j)} - v_{\max}^{(j)}\right)$ is negative. According to our rules for designing the individual contribution measure, all correct predictions make positive contributions. Thus a term $v_{\max}^{(j)}$ is added to $v_{\mathrm{sec}}^{(j)} - v_{\max}^{(j)}$ to normalize it to always be positive, which gives equation (6). And $v_{\max}^{(j)}$ is added to $\left(v_{\max}^{(j)} - v_{c_i(x_j)}^{(j)}\right)$ to maintain their relative order, which gives equation (5).

When $c_i(x_j)$ does not equal y_j (the third and the fourth subsets), $\mathrm{IC}_{(j)}^{i}$ is defined as:

$$\mathrm{IC}_i^{(j)} = v_{\mathrm{correct}}^{(j)} - v_{c_i(x_j)}^{(j)} - v_{\max}^{(j)}, \quad (7)$$

where $v_{\mathrm{correct}}^{(j)}$ is the number of votes for the correct label of d_j. The two negative cases can be considered together. Similar to the discussion of "degree of positive contribution", the "degree of negative contribution" is estimated by $\left(v_{\mathrm{correct}}^{(j)} - v_{c_i(x_j)}^{(j)}\right)$ which is the difference between the number of votes on the correct label and the number of votes on $c_i(x_j)$. The expression $\left(v_{\mathrm{correct}}^{(j)} - v_{c_i(x_j)}^{(j)}\right)$ could give a positive value, but according to our designing rules incorrect predictions should make negative contributions. So a term $\left(-v_{\max}^{(j)}\right)$ is added to $\left(v_{\mathrm{correct}}^{(j)} - v_{c_i(x_j)}^{(j)}\right)$ to normalize it to always be negative.

Combining equations (5), (6), and (7) with equation (4), the individual contribution of the classifier c_i is:

$$\mathrm{IC}_i = \sum_{j=1}^{N}\left(\alpha_{ij}\left(2v_{\max}^{(j)} - v_{c_i(x_j)}^{(j)}\right) + \beta_{ij}v_{\mathrm{sec}}^{(j)} + \theta_{ij}\left(v_{\mathrm{correct}}^{(j)} - v_{c_i(x_j)}^{(j)} - v_{\max}^{(j)}\right)\right), \quad (8)$$

where
$$\alpha_{ij} = \begin{cases} 1 \text{ if } c_i(x_j) = y_j \text{ and } c_i(x_j) \text{ is in the minority group;} \\ 0 \text{ otherwise.} \end{cases}$$

$$\beta_{ij} = \begin{cases} 1 \text{ if } c_i(x_j) = y_j \text{ and } c_i(x_j) \text{ is in the majority group;} \\ 0 \text{ otherwise.} \end{cases}$$

$$\theta_{ij} = \begin{cases} 1 \text{ if } c_i(x_j) \neq y_j; \\ 0 \text{ otherwise.} \end{cases}$$

Accordance to equation (8) it is forming the set of NN modules which include into ancemble. It demands the NNs modules combining.

VI. Procedure of NN Modules Combining

The output of an ensemble is a weighted average of the outputs of each network, with the ensemble weights determined as a function of the relative error of each network determined in training [14] – [16]; the resulting network often outperforms the constituent networks. There is a growing body of research into ensemble methods, for example, improvements in performance can result from training the individual networks to be decorrelated with each other [12], [13], [17], [18] with respect to their errors.

It used an approach to determine the ensemble weights dynamically as part of the training algorithm, i.e. during each propagation through the network, as opposed to any pre-determined fixed values or calculations [21]. The weights are proportional to the certainty of the respective outputs. The certainty of a network output measures how close the output is to one or the other of the target values.

We define the dynamically averaged network (DAN) by:

$$f_{\mathrm{DAN}} = \sum_{i=n}^{n} w_i f_i(x),$$

where the w_i ($i = \overline{1,n}$) are according to:

$$w_i = \frac{c\left(f_i(x)\right)}{\sum_{i=n}^{n} c\left(f_i(x)\right)},$$

where $c(f_i(x)) = \begin{cases} f_i(x) & \text{if } f_i(x) \geq 0.5, \\ 1 - f_i(x) & \text{otherwise.} \end{cases}$

The w is sum to 1, so f_{DAN} is a weighted average (WA) of the network outputs. The difference is that the weight vector is recomputed each time the ensemble output is evaluated, to try to give the best decision for the particular instance under consideration, instead of statically choosing weights that give an optimal decision with respect to a cross validation set. Each network's contribution to the sum is proportional to its certainty. A value close to 0.5, for instance, would contribute very little to the sum while a very certain value of 0.99 (or 0.01) among many less certain values would dominate the sum.

VII. Ensemble Pruning

Two classifiers are said to be complementary when their errors are uncorrelated. When complementary classifiers are combined in an ensemble, correct decisions are amplified by the aggregation process [10], [11].

The pruning strategies proposed are based on modifying the order of aggregation in the bagging ensemble: classifiers that are expected to perform better when combined are aggregated first. From the subensemble S_{u-1} of size $u - 1$, the subensemble S_u of size u is constructed by incorporating a single classifier selected from the set E_T / S_{u-1}, which contains the classifiers from the original ensemble not included in S_{u-1}. This classifier is identified using a rule that attempts to optimize the performance of the augmented ensemble S_u. The original random order of the pool of classifiers $t = 1; 2; ...; T$ is replaced by an ordered sequence $s_1; s_2; ...; s_T$, where s_j is the original label (in the randomly ordered bagging ensemble) of the classifier that occupies the jth position in the newly ordered ensemble. The curves that trace the evolution of the error as a function of the number of classifiers included in the ordered ensemble generally exhibit a minimum at intermediate ensemble sizes. This minimum corresponds to subensembles whose misclassification rates are below the error of the complete bagging ensemble. In this manner, approximate solutions to the problem of identifying near optimal subensembles can be obtained in polynomial time.

Finally, depending on the desired amount of pruning, the first τ classifiers in the sequence are selected. If the goal is to improve accuracy, τ should correspond to the minimum in the test set. Determining the location of this minimum using information only from the training set is a difficult task because test and train minima can occur at different subensemble sizes. Nonetheless, the minimum observed in the ensemble test error curves is fairly broad, which means that it is easy to improve the results of bagging by early stopping in the aggregation process in the ordered bagging ensembles.

VIII. Conclusion

It is proposed a new structure of NN ensemble, which elements are NN modules, that permits to increase its accuracy. It is considered a general approach of NN ensembles building: choice of kind and number of elements, procedure of NN modules combining and ensemble pruning. It is determined the best algorithms which supply the best solution of each task.

References

[1] G. E. Hinton, *A practical guide to training restricted Boltzmann machines*, (Tech. Rep. 2010-000). Toronto: Machine Learning Group, University of Toronto. 2010, pp. 160–169.

[2] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A niched Pareto genetic algorithm for multiobjective optimization," in *Proceedings of the First IEEE Conference on Evolutionary Computation*, vol. 1, Piscataway, 1994, pp. 82–87.

[3] Chumachenko E. I. Features of hybrid neural networks use with input data of different types / E. I. Chumachenko, D. Yu. Koval, G. A. Sipakov, D. D. Shevchuk // Electronics and Control Systems, N 4(42) – Kyiv: NAU, 2014. – pp. 91–97.

[4] Chumachenko O. I., Kryvenko I. V. Neural networks module learning // Electronics and Control Systems, N 2(48) – Kyiv: NAU, 2016. – pp. 76–80.

[5] L. Breiman. Bagging predictors. Machine Learning, 24(2):123–140, 1996.

[6] L. Breiman. Random forests. Machine Learning, 45(1):5 –32, 2001.

[7] W. Fan, H. Wang, P. S. Yu, and S. Ma. Is random model better? on its accuracy and efficiency. In Proceedings of the 3rd IEEE International Conference on Data Mining, pages 51 – 58, 2003.

[8] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1):119–139, 1997.

[9] T. K. Ho. The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(8):832–844, 1998.

[10] L.K. Hansen and P. Salamon, "Neural Network Ensembles," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 12, pp. 993-1001, 1990.

[11] A. Krogh and J. Vedelsby, "Neural Network Ensembles, Cross Validation, and Active Learning," Advances in Neural Information Processing Systems, G. Tesauro, D. Touretzky, and T. Leen, eds., vol. 7, pp. 231-238, MIT Press, 1995.

[12] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.

[13] Y. Freund and R.E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," Proc. Second European Conf. Computational Learning Theory, pp. 23-37, 1995.

[14] E. Bauer and R. Kohavi, "An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants," Machine Learning, vol. 36, nos. 1-2, pp. 105-139, 1999.

[15] G. Martrınez-Mu˜noz and A. Suarrez, "Aggregation Ordering in Bagging," Proc. IASTED Int'l Conf. Artificial Intelligence and Applications, pp. 258-263, 2004.

[16] G. Martrınez-Mu˜noz and A. Suŕarez, "Using Boosting to Prune Bagging Ensembles," Pattern Recognition Letters, vol. 28, no. 1, pp. 156-165, 2007.

[17] Y. Zhang, S. Burer, and W.N. Street, "Ensemble Pruning via Semi-Definite Programming," J. Machine Learning Research, vol. 7, pp. 1315-1338, 2006.

[18] G. Tsoumakas, L. Angelis, and I. Vlahavas, "Selective Fusion of Heterogeneous Classifiers," Intelligent Data Analysis, vol. 9, pp. 511-525, 2005.

[19] R.E. Banfield, L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer, "A Comparison of Decision Tree Ensemble Creation Techniques," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 29, no. 1, pp. 173–180, Jan. 2007.

[20] Zhenyu Lu, Xindong Wu+, Xingquan Zhu@, Josh Bongard, "Ensemble Pruning via Individual Contribution Ordering," http://www.cs.uvm.edu/~jbongard/papers/2010_KDD_Lu.pdf

[21] Imran Maqsood, Muhammad Riaz Khan, and Ajith Abraham, "An ensemble of neural networks for weather forecasting," *Neural Comput & Applic* (2004) 13: 112–122, DOI 10.1007/s00521-004-0413-4.

Received September 28, 2017

Chumachenko Olena. Candidate of Science (Engineering). Assosiate Professor.
Technical Cybernetic Department, National Technical University of Ukraine "Ihor Sikorsky Kyiv Polytechnic Institute," Kyiv, Ukraine.
Education: Georgian Politechnic Institute, Tbilisi, Georgia, (1980).
Research area: system analysis, artificial neuron networks.
Publications: more than 60 papers.
E-mail: chumachenko@tk.kpi.ua

Kot Anatoly. Post-graduate student.
Technical Cybernetic Department, National Technical University of Ukraine "Ihor Sikorsky Kyiv Polytechnic Institute," Kyiv, Ukraine.
Education: National Technical University of Ukraine "Ihor Sikorsky Kyiv Polytechnic Institute", (2017).
Research area: artificial Intelligence.
E-mail: anatoly.kot@gmail.com

О. І. Чумаченко, А. Т. Кот. Структурно-параметричний синтез ансамблів гібридних нейронних мереж

У статті розглянуто підхід до дизайну ансамблю нейронних мереж як колекції кінцевого числа нейронних мереж для вирішення однієї і тієї ж задачі, а потім об'єднання результатів їх роботи. Запропоновано алгоритм оптимального вибору топологій нейронних мереж і їх кількості для включення в ансамбль. Подальше уточнення складу ансамблю здійснюється за допомогою операції обрізання. Вихід ансамблю є середньозваженим значенням виходів кожної мережі, при цьому сукупні ваги визначаються як функція відносної похибки кожної мережі, визначеної при навчанні. Представлено новий підхід до динамічного визначення ансамблевих ваг в рамках алгоритму навчання. Ваги пропорційні визначеності відповідних виходів.

Ключові слова: нейронні мережі; ансамбль; навчання; оптимізація; топологія.

Чумаченко Олена Іллівна. Кандидат технічних наук. Доцент.
Кафедра технічної кібернетики, Національний технічний університет України «Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна.
Освіта: Грузинський політехнічний інститут, Тбілісі, Грузія, (1980).
Напрям наукової діяльності: системний аналіз, штучні нейронні мережі.
Кількість публікацій: більше 60 наукових робіт.
E-mail: chumachenko@tk.kpi.ua

Кот Анатолій Тарасович. Аспірант.
Національний технічний університет України «Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна.
Освіта: Національний технічний університет України «Київський політехнічний інститут ім. Ігоря Сікорського», Київ, Україна, (2017).
Напрям наукової діяльності: штучний інтелект.
E-mail: anatoly.kot@gmail.com

Е. И. Чумаченко, А. Т. Кот. Структурно-параметрический синтез ансамблей гибридных нейронных сетей

В статье рассмотрен подход к дизайну ансамбля нейронных сетей как коллекции конечного числа нейронных сетей для решения одной и той же задачи, а затем объединения результатов их работы. Предложен алгоритм оптимального выбора топологий нейронных сетей и их количества для включения в ансамбль. Дальнейшее уточнение состава ансамбля осуществляется с помощью операции обрезки. Выход ансамбля представляет собой средневзвешенное значение выходов каждой сети, при этом совокупные веса определяются как функция относительной погрешности каждой сети, определенной при обучении. Представлен новый подход к динамическому определению ансамблевых весов в рамках алгоритма обучения. Веса пропорциональны определенности соответствующих выходов.

Ключевые слова: нейронные сети; ансамбль; обучение; оптимизация; топология.

Чумаченко Елена Ильинична. Кандидат технических наук. Доцент.
Кафедра технической кибернетики, Национальный технический университет Украины «Киевский политехнический институт им. Игоря Сикорского», Киев, Украина.
Образование: Грузинский политехнический институт, Тбилиси, Грузия, (1980).
Направление научной деятельности: системный анализ, искусственные нейронные сети.
Количество публикаций: более 60 научных работ.
E-mail: chumachenko@tk.kpi.ua

Кот Анатолий Тарасович. Аспирант.
Национальный технический университет Украины «Киевский политехнический институт им. Игоря Сикорского», Киев, Украина.
Образование: Национальный технический университет Украины «Киевский политехнический институт им. Игоря Сикорского», Киев, Украина, (2017).
Направление научной деятельности: искусственный интеллект.
E-mail: anatoly.kot@gmail.com