Improving the architecture of a two-level heterogeneous ensemble for solving machine learning problems

Irina Kalinina[†], Aleksandr Gozhyj[*,†], Victor Gozhyi[†] and Sergii Shiyan[†]

Petro Mohyla Black Sea National University, St. 68 Desantnykiv 10, Mykolaiv, 54000, Ukraine

Abstract

The article investigates an approach to improving the architectures of two-level heterogeneous ensembles of models for solving machine learning problems. An improved ensemble architecture is proposed. In which the boosting method is used at the first level of ensemble learning to gradually improve the solutions of the base models. At the second level, the stacking method is used to aggregate the solutions of the base models using a metamodel. The base models used were a model based on multiple linear regression, a decision tree model, a random forest model, a support vector model, a KNN model, a model based on an artificial neural network, and a multivariate adaptive regression spline model. These models are divided into two groups: undertrained and over trained. The experimental part of the study was carried out on solving the problem of predicting the electricity generation indicators of hybrid power plants based on environmental indicators. The use of the improved architecture of a two-level heterogeneous ensemble demonstrated an increase in forecast accuracy compared to other ensemble architects and solutions based on any of the base models. The proposed approach is effective in solving machine learning problems.

Keywords

two-level heterogeneous ensemble of models, boosting, bagging, stacking, machine learning

1. Introduction

The use of ensemble models to solve machine learning problems has become a leading trend in recent years. The ensemble approach allows combining several weak models to form a strong model in order to improve the accuracy of solving machine learning problems. Combining uncorrelated predictions obtained using different alternative base models can improve the performance of the models and demonstrate a reduction in the overall error. Ensemble methods are designed to reduce bias or variance by aggregating the forecast values of the base (weak) models.

By combining the strengths of different base models, ensemble learning can identify more complex patterns and compensate for the weaknesses of individual models. This factor has led to increased accuracy of predictions with real-time data. Such ensemble methods can increase accuracy, reduce errors, and provide more reliable predictions by combining the power of different algorithms. This works especially well in complex cases where a single model would be too difficult to detect all the significant patterns present in the data.

Ensemble approaches are now being used in industries such as healthcare, finance, ecology, and cybersecurity to solve complex problems. As these industries generate vast amounts of data,

ISW-2025: Intelligent Systems Workshop at 9th International Conference on Computational Linguistics and Intelligent Systems (CoLInS-2025), May 15–16, 2025, Kharkiv, Ukraine

[*] Corresponding author.

[†] These authors contributed equally.

irina.kalinina1612@gmail.com (I. Kalinina); alex.gozhyj@gmail.com (A. Gozhyj); gozhyi.v@gmail.com (V. Gozhyi); shyiansi@gmail.com (S. Shiyan)

0000-0001-8359-2045 (I. Kalinina); 0000-0002-3517-580X (A. Gozhyj); 0000–0002–5341–0973 (V. Gozhyi); 0000-0001-9255-9511 (S. Shiyan)

requiring more sophisticated analysis than ever before, ensemble modeling has become essential to driving innovation. For example, in healthcare, ensemble models can help combine analytical data from multiple sources and models to identify disease risks, optimize treatment programs, and accelerate drug discovery. Therefore, the more sources of information that are combined, the better the outcome in many industries.

Ensemble methods can be divided into two groups: homogeneous and heterogeneous ensembles. Homogeneous ensembles are formed by aggregating models of the same type to solve machine learning problems. Heterogeneous ensembles are of particular interest. They are formed by aggregating predictions obtained using different types of models. Their application allows for a significant and comprehensive reduction in the overall error in solving a machine learning problem.

Problem statement. To investigate the features of the process of building two-level heterogeneous ensembles of models. To develop an improved architecture of heterogeneous ensembles of models to reduce the overall error of solving machine learning problems. To experimentally confirm the effectiveness of the improved architecture of a two-level heterogeneous ensemble of models on the example of solving a forecasting problem.

2. Related Works

Ensemble learning is the process of aggregating several different models that solve machine learning problems to obtain results that are better than those obtained by the algorithms when used independently. Ensemble methods are often used to obtain solutions to various machine learning problems. First of all, these are classification and prediction problems [1-3].

The main principles of ensemble learning are presented in [4]. To increase the efficiency of solving machine learning problems, ensemble learning takes advantage of several base models. In [5], the authors present results showing that using ensembles of models yields more accurate results than individual machine learning models. The work also shows that ensemble classifiers outperform individual ensemble learning models and are more reliable.

Individual base models used in solving machine learning problems have high variance and high bias, which affects the overall accuracy of predictions [6,7]. Finding a compromise between variance and bias allows you to increase the accuracy of predictive solutions. The variance and bias errors caused by individual machine learning models can be reduced using ensemble methods, for example, bagging reduces variance without increasing bias, and boosting reduces bias [8,9]. The boosting algorithm allows for successively improving the solutions of several alternative base models. The specifics of its use are presented in detail in [10].

There are two types of approaches to ensemble learning: parallel fitting of several alternative models and sequential improvement of the solutions of these methods. The parallel method trains different base models separately and then combine their solutions, sometimes using a metamodel. This approach is implemented by bagging and stacking methods. Bagging, in the form of a random forest, is popular and is used in various projects [8,11]. Sequential ensemble models are trained sequentially so that each model learns to correct the error made by the previous model. In [12], it was determined that the accuracy of each base model is an important factor in the effectiveness of ensemble learning. Any machine learning algorithm is considered effective only when it has an effective generalization strategy on previously unstudied examples. Therefore, by combining the capabilities of many base models and approaches, ensemble learning is used to improve both the accuracy and the solution efficiency of the overall machine learning model.

A comprehensive approach to reducing the overall error in machine learning problems is implemented by using two-level heterogeneous ensembles. At one of the levels in such ensembles, the stacking method is used [13]. Stacking uses parallel training of several alternative models to aggregate the predictions of several members of the ensemble. The stacking procedure involves the use of several level 0 models as baselines and the meta-training procedure, which is a strategy that instructs another model to add predictions using the baseline models. In [13], a level 1 model is used. The main idea behind stacking is that baseline (level 0) models are trained using a training dataset. The input and output pairs of the new dataset are then used for training and meta-training by combining their expected target labeling with real labels on the hidden data [14].

Using the results of additional machine learning algorithms, meta-learning algorithms are trained to produce predictions that are more accurate than those generated by other base classifiers [15]. The stacking method is effective because it combines the advantages of many weak models to provide a result that is superior to that produced by ensemble models. In this case, many base algorithms are used, as well as an initial data set. This allows stacking to create unique models that solve the prediction problem in a new way.

The main difficulty in creating heterogeneous ensembles is determining the optimal way to combine the predictions of different models in the ensemble. There are two ways to build a heterogeneous ensemble. In the first method, a fixed number of different models are combined. The second method is to create a group of models with different parameters, and then select the best subset to include in the final ensemble.

In [16], the issues of creating an adaptive heterogeneous ensemble for solving machine learning problems are considered. In [17], the authors proposed a static heterogeneous ensemble architecture. When creating an ensemble, 5 different basic classifiers are combined: the support vector machine (SVM), the multilayer perceptron (MLP), logistic regression, the K-nearest neighbors' method, and a decision tree. The parameters and architecture of individual classifiers are determined using 10-fold cross-validation. The proposed approach is effective in solving classification problems.

The authors of [18] proposed a combination of several optimized methods: deep neural networks, SVM, ada-boost, and Gaussian processes. The ensemble is generated using a simple sum-of-classifiers rule. However, the problem of determining the number of classifiers of each type was not considered. In addition, the optimal composition of the ensemble depends on the problem being solved. In [16–18], a possible way to overcome this problem was also proposed. It consists of creating a library of classifiers and then selecting a subset for the final ensemble.

In [19], a library of 2000 different methods trained with a wide range of different parameters is built. From this library of models, an iterative greedy selection algorithm is applied to build a final ensemble. The procedure starts with an empty ensemble. Then, at each iteration, the model that maximizes the performance metric is included in the ensemble until all models are in the library. The ensemble with the best performance in the validation set is selected as the final combination.

In [20], the authors propose a greedy selection method from a library consisting of 200 classifiers: 40 neural networks, 60 nearest-neighbor classifiers, 80 SVMs, and 20 decision trees. For each type of classifier, a parameter grid was defined, and one model was trained for each node in the grid. In this approach, the ensemble grows gradually, selecting one classifier from the library at a time. At each step, the selection is made from the perspective of both individual accuracy and complementarity with the rest of the classifiers in the ensemble. In the classification problems studied, such heterogeneous ensembles turned out to be more accurate.

In [21,22], a genetic algorithm was considered for selecting the optimal structure of a heterogeneous ensemble from 20 different base models. These selection methods have been widely applied to homogeneous ensembles. Examples are presented in [23, 24]. In [25], to build an efficient heterogeneous combination, the authors remove the low-performing base elements so that only the optimal classifiers remain in the ensemble. The efficiency of the classifier is determined by measuring the area under the ROC curve. In another study [26], the authors used a differential evolution algorithm to optimize the weights of various base models in a heterogeneous ensemble.

Thus, the task of improving the architecture of a heterogeneous ensemble in order to improve the quality of predictive solutions is complex and relevant. It requires new approaches and additional research.

3. Material and methods

One of the approaches to improving the quality of results when solving machine learning problems is the use of ensemble learning. This approach allows you to reduce the errors of solving problems by gradually reducing the bias and variance. Studies have shown that it is most effective to use a multi-level heterogeneous combination of different ensemble methods based on bagging, boosting and stacking [2].

It is known that the error of machine learning algorithms consists of three components: noise, bias, and variance [1,3]:

$$Error(x) = Bias(x)^2 + Variance(x) + Noise(x), \tag{1}$$

where $Bias$ — is a systematic error that any machine learning algorithm is expected to make due to, for example, the choice of model structure, insufficient amount of training data, unrepresentative data for training the model; $Variance$ measures the sensitivity of the algorithm to a specific training set and/or selected hyperparameters; $Noise$ — This is a random error in the data that cannot be avoided, for example, due to data entry errors, previously corrupted input data, etc.

When choosing the best base model for a machine learning problem, it is necessary to pay attention to obtaining a compromise between bias and variance because undertrained models have a large bias value, and over trained models have a large variance value. These shortcomings lead to an increase in the overall error. The technique of building a two-level architecture of the ensemble of models, that is, aggregating solutions from different base models to create a single generalized model with a smaller error value, is a technique that can find a compromise between the values of bias and variance of individual base models. Thus, a two-level ensemble architecture can help reduce both bias and variance, excluding the noise component of the error.

When selecting a combination of ensemble methods and forming the architecture of a heterogeneous ensemble, the following features should be considered:

- combining the results of several base models reduces the risk of choosing an ineffective (weak) model;
- several uncorrelated models, grouped into an ensemble, provide a more accurate solution than any of the individual base machine learning models;
- ensemble methods tend to improve the generalized accuracy of only a set of individual models, and all this only happens in a certain domain;
- a set of base machine learning models with similar training results may have different generalization results;
- if the initial data set is too large, then one model may not cope with the solution of the problem. In this case, it is necessary to train different base models on different data samples;
- if the initial data set is too small, then resampling methods should be used.

Let us define the relationship between error, bias, and variance. Let us have random variables — $X, Y, \hat{Y}$, describing the distribution of values for instances *x*; their real *f(x)* and predicted values *h(x)*. The value *h(X)* is an estimate of the true function *f(X)*, generated by some model M_i, but is unknown.

When constructing any process model, we assume that the observed values $y \in Y$ are generated by the function $f(X)$ plus a random normally distributed error, ε:

$$Y = f(x) + \varepsilon.$$

The mean squared error that we expect for the entire data distribution is defined as follows:

$$MSE = E[(h(X) - f(X))^2] = E[(\hat{Y} - f(X))^2]. \tag{2}$$

Let us define the components of error: bias and variance. Bias is the difference between the mean values determined by the model and the actual values. Bias shows how much the “average” model differs from the actual relationship between the variables (Fig. 1). It can be represented as:

$$Bias^2(h, f) = (E[h(X)] - f(X))^2 = E^2[h(X)] + f(X)^2 - 2E[h(X)]f(X).$$

Variance is the expected variability of a model around its mean. Variance shows how much the model changes, for example with different hyperparameters or data samples (Fig. 1). It can be represented as:

$$Var(h) = E[(h(X) - E[h(X)]^2].$$

Substitute the defined expressions for the bias and dispersion into expression (2) and obtain the following dependence:

$$MSE = E[(h(X) - f(X))^2] = E[(h(X) + E[h(x)] - E[(h(X) - f(X)])^2] = \cdots$$
$$\ldots = \left(E[h(X)] - f(X)\right)^2 + E\left[\left(h(X) - f(X)\right)^2\right] = \left(E(\hat{Y}) - Y\right)^2 + E\left[\left(\hat{Y} - Y\right)^2\right] + \varepsilon =$$
$$= Bias^2 + Variance + irreducible\ error.$$

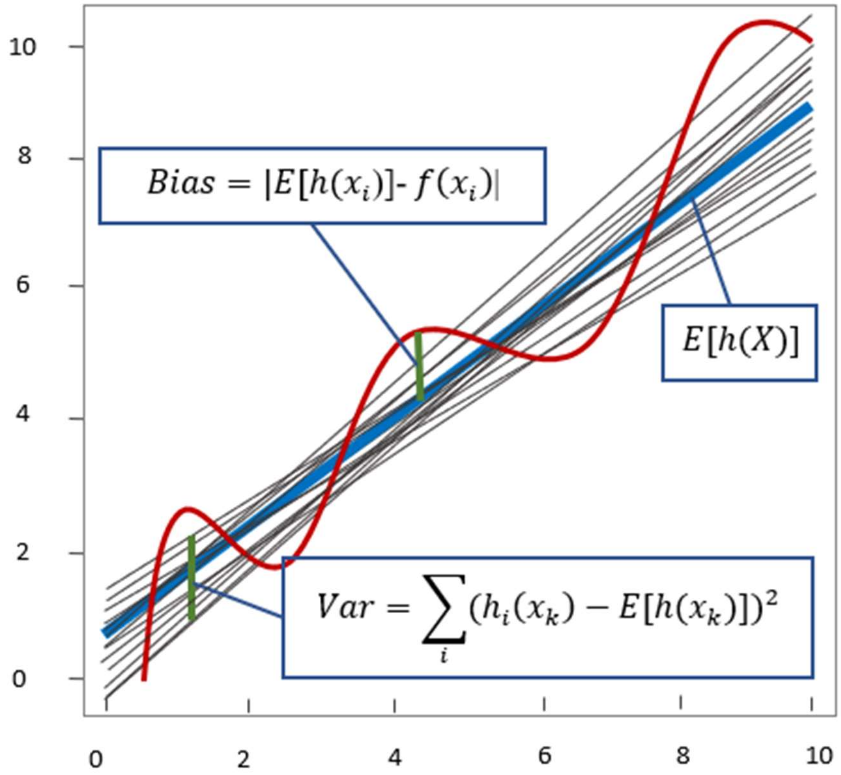


Figure 1: The relationship between bias and variance in model fitting

By combining several alternative and uncorrelated solutions to a machine learning problem into an ensemble structure, the variance is usually reduced, and therefore the error rate is reduced.

$$Var(Ensamble(h_i(x, D))) = \frac{\sum_m Var(h_i(x, D_i))}{m},$$

where $h_i(x, D_i)$ – the predicted value obtained using the i-th model for training from the ensemble of models that is formed.

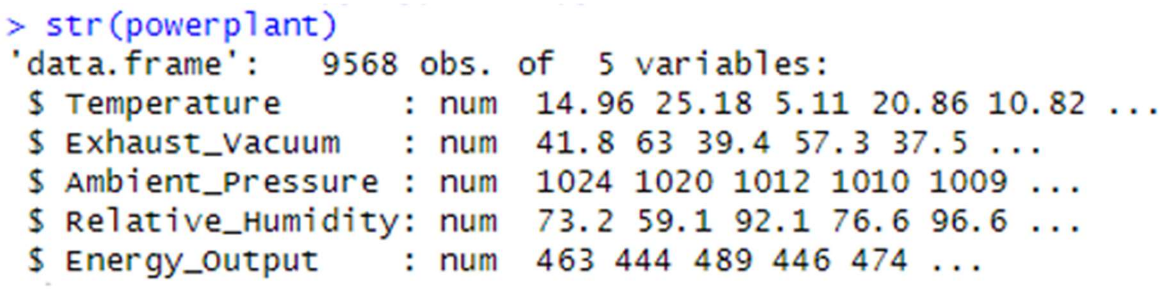
The use of ensembles of models contributes to the fact that for most cases the variance of the ensemble of models is usually lower than in the case of using a single base model, even if it is complex.

4. Experimental part

4.1. Data analysis and pre-processing

As an example of the application of the ensemble approach, the task of predicting the electricity generation indicators of hybrid power plants based on environmental indicators [27] is considered. The task is to predict the electricity generation of a combined cycle plant based on environmental indicators collected using a system of sensors located near the power plant. Thus, the data set formed on the basis of sensor indicators was obtained from actual observations. The data set presents 9568 observations collected over 6 years (2006–2011). At the same time, full load in the combined cycle mode was for 674 days.

The result of the study is the amount of electricity generation (PE 420.26 – 495.76 MW), which is taken as the target variable. The input data are ambient temperature (AT in the range from 1.81◦C to 37.11◦C), ambient pressure (AP in the range from 992.89 – 1033.30 millibars) and relative humidity (RH in the range from 25.56% to 100.16%). The dataset consists of numerical features. The data does not require additional processing. The structure of the dataset is presented in Figure 2.



```
> str(powerplant)
'data.frame':   9568 obs. of  5 variables:
 $ Temperature      : num  14.96 25.18 5.11 20.86 10.82 ...
 $ Exhaust_Vacuum   : num  41.8 63 39.4 57.3 37.5 ...
 $ Ambient_Pressure : num  1024 1020 1012 1010 1009 ...
 $ Relative_Humidity: num  73.2 59.1 92.1 76.6 96.6 ...
 $ Energy_Output    : num  463 444 489 446 474 ...
```

Figure 2: Dataset structure

The data analysis and pre-processing stage includes the following procedures: identification and processing of missing values, identification and processing of outliers and anomalous values, identification of duplicates, checking for correlation in the data, normalization and feature selection [28,29].

After checking for missing values and identifying anomalies, a correlation check was performed on the data. For this purpose, a correlation matrix and a scatter matrix were created. A graphical version of the correlation matrix is shown in Fig. 3.

Analysis of the correlation matrix shows:

1. Temperature and exhaust vacuum have a strong positive correlation of 0.84, which means that an increase in temperature is usually accompanied by an increase in exhaust vacuum.
2. Temperature and output power have a strong negative correlation of -0.95, which means that as the ambient temperature increases, the output power decreases.
3. Exhaust vacuum and power output have a strong negative correlation of -0.87, which indicates that if the exhaust vacuum is high, the power output is very low, and vice versa.

Duplicates (82 values) that were detected were removed. To increase the speed of training, data normalization was performed using the min-max method. Normalization transformed the features into the range [0, 1], which eliminated the differences between the scales [30].

After the dataset was prepared and reduced to a single range, cross-validation was performed to select the features to determine the best model with the corresponding important features. To evaluate the accuracy of the model using cross-validation on the sample, the MSE calculation method was used. First, the model characteristics were selected, then the errors were calculated for each cross-validation block. The result of feature selection demonstrated that the best model includes all 4 predictors. The errors of all models are very low, which is a good indicator.

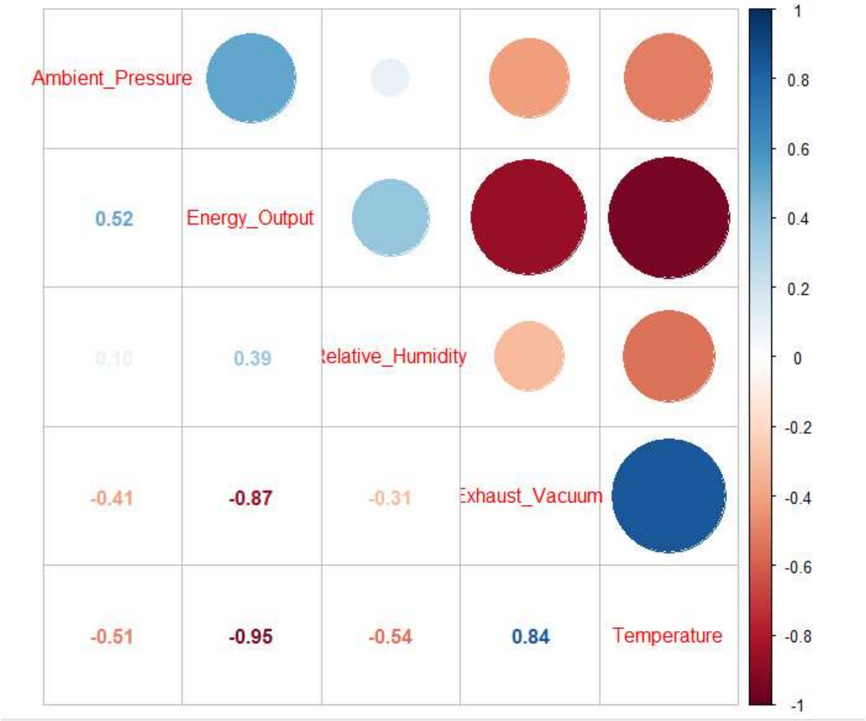


Figure 3: Graphical version of the correlation matrix

4.2. Building and training basic regression models

The first step in the modelling phase is to divide the dataset into training and testing. For the training sample, 90% of the rows of the cleaned dataset were selected, which were intended for training the basic predictive models. The remainder, which is 10% of the rows, was used for testing the models. The basic predictive models considered were a regression model based on multiple linear regression, a decision tree model, a random forest regression model, a support vector regression model, a KNN regression model, a regression model based on an artificial neural network, and a multidimensional adaptive regression spline model. For each model, a structure was selected and parameters were found, at which the models had the best quality indicators of predictions on the test data sample. Table 1 presents the values of the quality metrics after training and testing each of the basic regression models.

Table 1
Quality metrics of basic predictive models

No.	Model type	R^2	RMSE	MAE	MAPE %
1	Regression model based on MLR	0.94	3.51	2.56	0.84
2	Regression model based on DR	0.89	4.18	3.41	1.02
3	Regression model based on RF	0.96	3.39	2.46	0.53
4	Regression model based on SVM	0.95	3.48	2.54	0.64
5	Regression model based on KNN	0.91	3.77	2.89	0.93
6	Regression model based on ANN	0.90	3. 96	3.19	0.96
7	Multivariate adaptive regression splines (MARS)	0.95	3.45	2.51	0.59

So, the basic models showed good results on this data set. The random forest model performed best. The regression model based on the decision tree and the regression model based on the artificial neural network showed the worst results.

4.3. Formation of a two-level ensemble architecture based on basic models

Initial version. One effective approach to building heterogeneous ensemble architectures uses boosting or stacking for undertrained models and bagging for over trained base models [31,32]. This approach is aimed at minimizing bias in models with low variance and high bias, as well as reducing variance in models with high variance and low bias. This combination option was implemented programmatically to evaluate the effectiveness of this approach in the problem of regression forecasting the amount of electrical energy generated in one hour by a combined cycle power plant.

To form the initial variant of combining the basic regression models into a two-level heterogeneous ensemble architecture, it was decided to combine the undertrained models at the first level, and the over trained ones at the second level. To divide the basic models into two groups, the bias and variance values for each basic model were analysed. Thus, a regression model based on an artificial neural network and a decision tree were selected for the first level, and a regression model based on multiple linear regression, a regression model based on KNN, a regression model based on the support vector method, a regression model based on a random forest, and a multidimensional adaptive regression spline model were selected for the second level. The architecture of the initial variant of combining the basic regression models is presented by the stacking method at the first level and the bagging method at the second level (Fig. 4). A control scheme with cross-validation was used to configure the stacking. Based on the predictions of the basic undertrained models, a metamodel is created using a generalized linear model. The results of the ensemble learning from the first level, stacking, were used to predict on the test dataset. The obtained predictions were added to the predictions of the base models of the second level, which included the retrained models. The ensemble learning scheme based on bagging with the use of cross-validation was programmatically configured.

To assess the effectiveness of ensemble learning, the quality indicators of models and forecasts were calculated: RMSE, MAE, MAPE and R^2. The results of the initial version of combining the basic models into a two-level heterogeneous ensemble architecture are presented in Table 2.

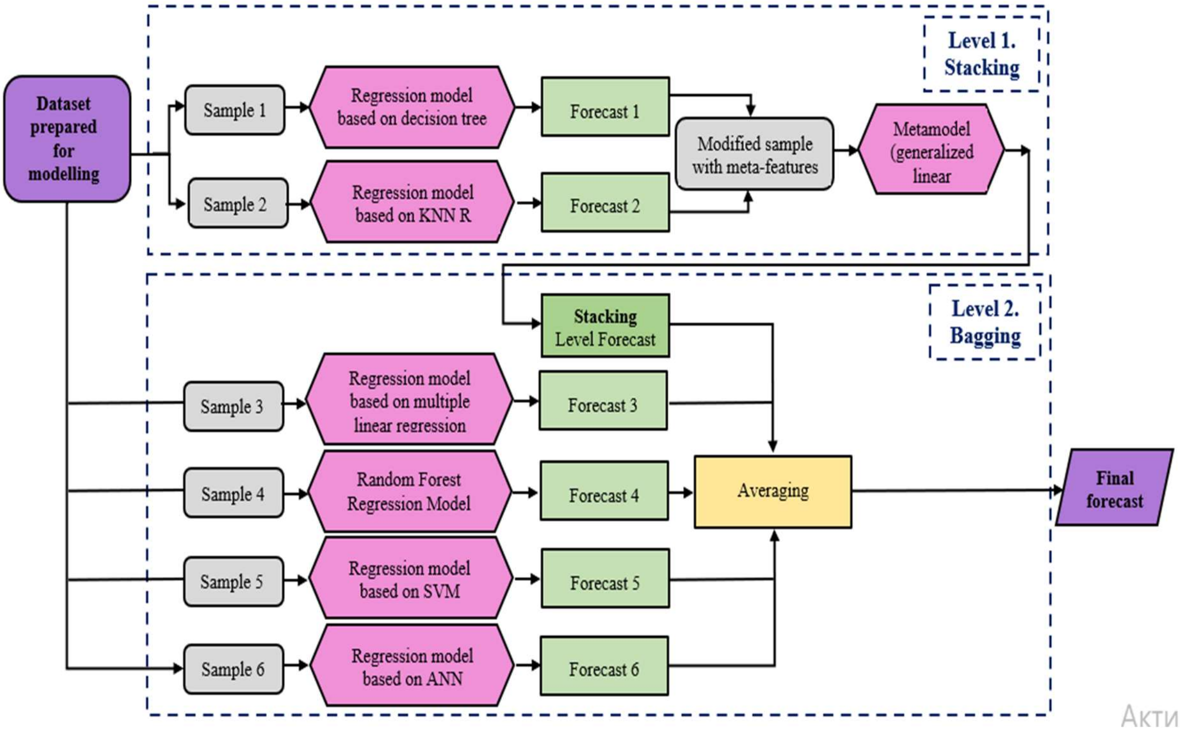

Level 1.
Stacking
Dataset prepared for modelling
Sample 1
Regression model based on decision tree
Forecast 1
Sample 2
Regression model based on KNN R
Forecast 2
Modified sample with meta-features
Metamodel (generalized linear
Level 2.
Bagging
Stacking Level Forecast
Sample 3
Regression model based on multiple linear regression
Forecast 3
Sample 4
Random Forest Regression Model
Forecast 4
Averaging
Final forecast
Sample 5
Regression model based on SVM
Forecast 5
Sample 6
Regression model based on ANN
Forecast 6

Figure 4: The initial version of the combination of basic regression models into a two-level heterogeneous ensemble architecture

Table 2
Results of the initial version of combining basic regression models into a two-level heterogeneous ensemble architecture

No.	Model type	Model quality	Forecast quality		
		R^2	RMSE	MAE	MAPE %
1	Regression model based on ANN	0.90	3. 96	3.19	0.96
	Regression model based on DR	0.89	4.18	3.41	1.02
	Resulting stacking layer	**0.93**	**3.91**	**2.90**	**0.91**
2	Regression model based on MLR	0.94	3.51	2.56	0.84
	Regression model based on KNN	0.91	3.77	2.89	0.93
	Regression model based on SVM	0.95	3.48	2.54	0.64
	Regression model based on RF	0.96	3.39	2.46	0.53
	Multivariate adaptive regression splines (MARS)	0.95	3.45	2.51	0.59
Resulting bagging layer		**0.95**	**3.40**	**2.48**	**0.60**

At the first level, using stacking, it was possible to increase the accuracy of forecasts to 0.93, which is 4% more compared to individual baseline models, such as ANN and decision tree (with R^2 = 0.90 and 0.89).

At the second level, bagging combined the forecasts of the first level of stacking and retrained models, which allowed to achieve overall accuracy in forecasting. Contrary to this, a separate baseline regression model based on random forest outperformed the results of bagging. The reason for such results is the specificity of the task of regression forecasting the amount of electricity generated in one hour by a combined cycle power plant and the features of the data set as a whole.

Improved version. When forming an improved version of the ensemble architecture, an unconventional approach was chosen, which involves the use of undertrained models at the first level aggregated by the boosting method and over trained models at the second level aggregated by the stacking method (Fig. 5). This decision is due to the hypothesis that undertrained models, due to less specificity in data fitting, can more effectively capture general trends and reduce the risk of overtraining.

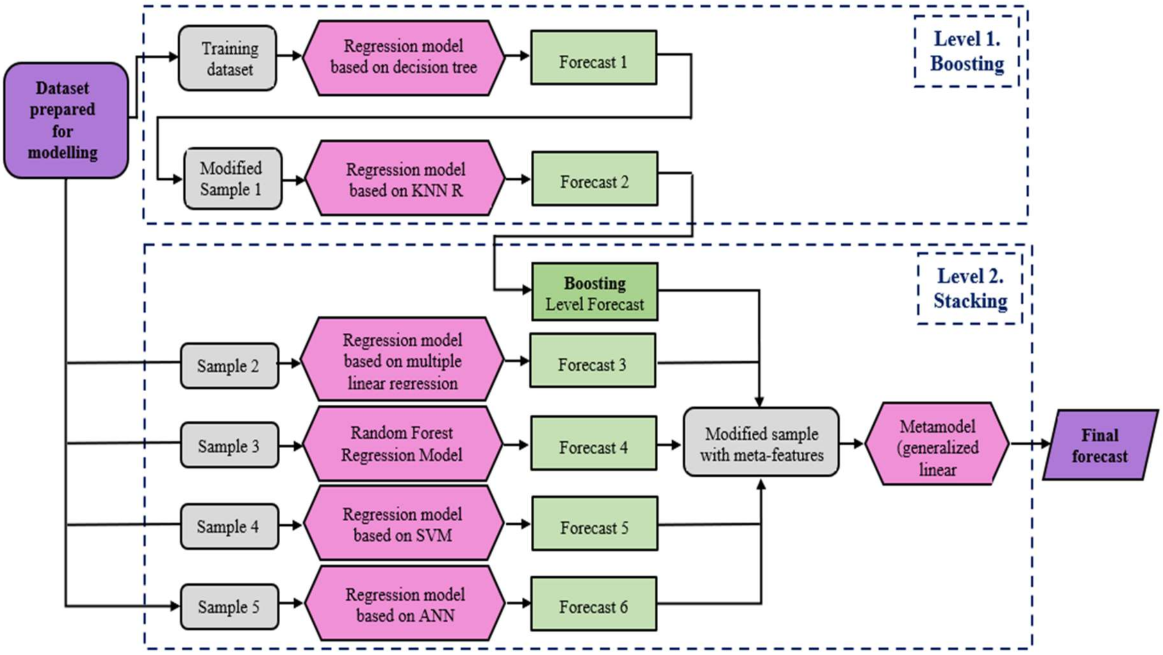

Dataset prepared for modelling
Training dataset
Regression model based on decision tree
Forecast 1
Level 1. Boosting
Modified Sample 1
Regression model based on KNN R
Forecast 2
Boosting Level Forecast
Level 2. Stacking
Sample 2
Regression model based on multiple linear regression
Forecast 3
Sample 3
Random Forest Regression Model
Forecast 4
Modified sample with meta-features
Metamodel (generalized linear
Final forecast
Sample 4
Regression model based on SVM
Forecast 5
Sample 5
Regression model based on ANN
Forecast 6

Figure 5: An improved version of the combination of basic regression models into a two-level heterogeneous ensemble architecture

The boosting method used at the first level allows to increase the stability of a heterogeneous ensemble by combining the results of several weakly correlated models. At the second level, the retrained base models and the boosting results are aggregated using stacking. This provides more accurate forecasting, because stacking uses the results of the initial forecasting for further training of the models, which allows them to deepen their knowledge of the dependencies in the data.

The obtained training results of the second layer, stacking, are used to predict the value of the output variable on the test data set. To assess the accuracy of the obtained forecasts, the main indicators such as RMSE, MAE, MAPE and the coefficient of determination R^2 are calculated. The results on the quality indicators of models and forecasts are summarized in Table 3.

At the first level, where boosting was used, the prediction accuracy increased to 0.91, compared to 0.89 and 0.90 for individual models (ANN and decision tree). At the second level, when stacking was used, an improvement in performance was achieved compared to the results of individual base models: R^2 = 0.97, RMSE = 3.37, MAE = 2.44, MAPE = 0.52%, which confirms the high efficiency of model combination. Thus, the improved two-level ensemble architecture exceeds the efficiency of individual models in prediction accuracy.

Table 3
Results of an improved version of combining basic regression models into a two-level heterogeneous ensemble architecture

No.	Model type	Model quality	Forecast quality		
		R^2	RMSE	MAE	MAPE %
1	Regression model based on ANN	0.90	3. 96	3.19	0.96
	Regression model based on DR	0.89	4.18	3.41	1.02
	Resulting boosting layer	**0.91**	**3.81**	**2.95**	**0.95**
2	Regression model based on MLR	0.94	3.51	2.56	0.84
	Regression model based on KNN	0.91	3.77	2.89	0.93
	Regression model based on SVM	0.95	3.48	2.54	0.64
	Regression model based on RF	0.96	3.39	2.46	0.53
	Multivariate adaptive regression splines (MARS)	0.95	3.45	2.51	0.59
Resulting stacking layer		**0.97**	**3.37**	**2.44**	**0.52**

This variant of ensemble combination worked particularly effectively due to the specificity of the data set and the nature of the dependencies in it, as well as the features of the regression problem. The data set contains a variety of interdependencies between variables, which creates complex but significant patterns that need to be properly detected and taken into account for accurate construction of the regression model. Undertrained models, such as decision trees and KNN, are not powerful enough to reveal these dependencies individually, but combining them through boosting helps to better generalize the main, stable patterns in the data, without being prone to overtraining.

Over trained models, thanks to their processing algorithms, can better adapt to the complex nonlinear dependencies that are present in the data set, but are prone to fitting to the training data. Stacking helps to effectively combine their predictions, smoothing out the errors caused by overtraining and increasing the generalization ability of the ensemble on the test data.

Therefore, it is the specificity of the data, which has a combination of both stable and complex nonlinear dependencies, as well as the requirements of the regression problem, that justify the effectiveness of this combination of bagging for undertrained models and stacking for over trained ones. This approach allows to reveal significant relationships in the data and ensures the accuracy of predictions by better generalizing to new data.

5. Conclusions

The paper investigated an approach to improving ensemble architectures for solving machine learning problems. When choosing a combination of ensemble methods and forming a heterogeneous ensemble architecture, features that affect the construction of ensembles were identified. An improved ensemble architecture was developed. In which the boosting method is used at the first level of ensemble learning to gradually improve the solutions of basic models. At the second level, the stacking method is used to aggregate the solutions of basic models using a metamodel. In the experimental part of the study, the problem of predicting electricity generation indicators was solved. The basic models selected were a model based on multiple linear regression, a decision tree model, a random forest model, a support vector model, a KNN model, a model based on an artificial neural network, and a spline model of multivariate adaptive regression. These models were divided into undertrained and over trained groups. On the models of the first group, ensemble methods reduced the variance, and on the models of the second group, they reduced the bias. Thus, the forecasting results were improved by reducing the overall error. This made it possible to increase the forecast accuracy in the improved architecture of a two-level heterogeneous ensemble compared to other ensemble architects and solutions based on any of the basic models. The proposed approach is effective in solving machine learning problems.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] G. Kunapuli, Ensemble Methods for Machine Learning, United States of America: Manning, E-book (2023), p. 352.

[2] A. Kumar, M. Jain, Ensemble Learning for AI Developers: Learn Bagging, Stacking, and Boosting Methods with Use Cases, Germany: Apress, E-book (2020), p. 136.

[3] L. Rokach, Ensemble Learning: Ensemble Learning: Pattern Classification Using Ensemble Methods (Second Edition), Singapore: World Scientific Publishing Company, E-book (2019), p. 300.

[4] I. D. Mienye, Y. Sun, A survey of ensemble learning: Concepts, algorithms, applications, and prospects, IEEE Access, 10 (2022) pp. 99129–99149. doi: 10.1109/ACCESS.2022.3207287.

[5] J. Langford, A. D. Smith, T. J. Brown, Ensemble learning, in: C. Sammut, G. I. Webb (Eds.), Encyclopedia of Machine Learning, Springer, Boston, MA, (2011) pp. 312–320, doi: 10.1007/978-0-387-30164-8_252.

[6] S. Mishra, Y. Sun, A. Sharma, Improving the accuracy of ensemble machine learning classification models using a novel bit-fusion algorithm for healthcare AI systems, Frontiers in Public Health, 10 (2022) 858282, doi: 10.3389/fpubh.2022.858282.

[7] Y. Sun, Z. Li, X. Li, J. Zhang, Classifier selection and ensemble model for multiclass imbalance learning in education grants prediction, Applied Artificial Intelligence, vol. 35(3) (2021) pp. 1–14, doi: 10.1080/08839514.2021.1877481.

[8] S. Doroudi, The bias-variance tradeoff: How data science can inform educational debates, AERA Open, vol. 6(4) (2020) 233285842097720, doi: 10.1177/2332858420977208.

[9] S. Alelyani, Stable bagging feature selection on medical data, Journal of Big Data, vol. 8(1) (2021) 11. doi: 10.1186/s40537-020-00385-8.

[10] T. Ravichandran, S. Suresh, R. Kumar, Ensemble-based machine learning approach for improved leak detection in water mains, Journal of Hydroinformatics, vol. 23(2) (2021) 307–323. doi: 10.2166/hydro.2021.093.

[11] H. Liu, A. Gegov, M. Cocea, Ensemble learning approaches, in: Rule-Based Systems for Big Data, vol. 13 of Studies in Big Data, Springer, Cham, 2016, pp. 63–73. doi: 10.1007/978-3-319-23696-4_6.

[12] P. A. Flach, T. De Bie, N. Cristianini, Machine Learning and Knowledge Discovery in Databases: European conference, ECML PKDD 2012, vol. 7523–7524 of Lecture Notes in Artificial Intelligence, Springer, Berlin; New York, NY, 2012, URL: http://hdl.handle.net/1854/LU-7009607

[13] O. Sagi, L. Rokach, Ensemble learning: A survey, Wiley interdisciplinary reviews: data mining and knowledge discovery, vol. 8(4): e1249, (2018), doi: 10.1002/widm.1249.

[14] M. Liang, et al., A stacking ensemble learning framework for genomic prediction, Frontiers in Genetics, sec. Statistical genetics and methodology, vol. 12 (2021) 600040, doi: 10.3389/fgene.2021.600040.

[15] T. M. Hospedales, P. M. Antoniou, A. J. Storkey, Meta-learning in neural networks: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence, pp (99):1-1 (2021). doi: 10.1109/TPAMI.2021.3079209.

[16] Z. Lu, X. Wu, J. C. Bongard, Active learning through adaptive heterogeneous ensembling, IEEE Transactions on Knowledge and Data, vol. 27, issue 2 (2015) pp. 368–381, doi: 10.1109/TKDE.2014.2304474.

[17] J. M. de Oliveira, E. M. dos Santos, J. R. H. Carvalho, L. A. de Vasconcelos Marques, Ensemble of heterogeneous classifiers applied to lithofacies classification using logs from different wells, in conference: Neural Networks (IJCNN), International Joint, Dallas, TX, USA, (2013) pp. 1–6, doi: 10.1109/IJCNN.2013.6707013.

[18] L. Nanni, S. Brahnam, S. Ghidoni, A. Lumini, Toward a general-purpose heterogeneous ensemble for pattern classification, computational intelligence and neuroscience, vol. 5 (2015) pp. 1-10, doi: 10.1155/2015/909123.

[19] R. Caruana, A. Niculescu-Mizil, G. Crew, A. Ksikes, Ensemble selection from libraries of models, in: proceedings of the 21 st International Conference on Machine Learning, Banff, Canada, (ICML ’04), New York, USA, (2004) p. 18.

[20] I. Partalas, G. Tsoumakas, I. Vlahavas, An ensemble uncertainty aware measure for directed hill climbing ensemble pruning, Machine Learning, vol. 81(3) (2010) pp. 257–282, doi: 10.1007/s10994-010-5172-0.

[21] M. N. Haque, N. Noman, R. Berretta, P. Moscato, Heterogeneous ensemble combination search using genetic algorithm for class imbalanced data classification, PLoS One 11(1) (2016) e0146116, doi: 10.1371/journal.pone.0146116.

[22] G. Tsoumakas, I. Katakis, I. Vlahavas, Effective voting of heterogeneous classifiers, in conference: 15th European Conference on Machine Learning/8th, European conference on volume: 3201, Springer, Berlin, (2004) pp. 465–476, doi: 10.1007/978-3-540-30115-8_43.

[23] G. Tsoumakas, I. Partalas, I. Vlahavas, An ensemble pruning primer, in book: Applications of supervised and unsupervised ensemble methods, Springer, Berlin, (1970) pp. 1–13, doi: 10.1007/978-3-642-03999-7_1.

[24] G. Martínez-Muñoz, D. Hernández-Lobato, A. Suárez, An analysis of ensemble pruning techniques based on ordered aggregation, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31(2) (2009) pp. 245–259, doi: 10.1109/TPAMI.2008.78.

[25] E. Alshdaifat, M. Al-Hassan, A. Aloqaily, Effective heterogeneous ensemble classification: An alternative approach for selecting base classifiers, ICT Express, South Korea (2020), ICT Express 7(3), pp. 1-8, doi: 10.1016/j.icte.2020.11.005

[26] M. N. Haque, M. N. Noman, R. Berretta, P. Moscato, Optimising weights for heterogeneous ensemble of classifiers with differential evolution, in: Proc. IEEE Congress on Evolutionary Computation (CEC), Vancouver, BC, Canada. 2016, pp. 233–240, doi: 10.1109/CEC.2016.7743800.

[27] Machine learning repository [electronic resource], URL: https://archive.ics.uci.edu/ml/datasets/combined+cycle+power+plant

[28] I. Kalinina, A. Gozhyj, V. Vysotska, E. Malakhov, V. Gozhyj, I. Tregubova, System methodology of data analysis and preprocessing for solving classification problems, conference: 2024 IEEE 19th International Conference on Computer Science and Information Technologies (CSIT). Lviv, Ukraine (2024) doi:10.1109/CSIT65290.2024.10982630. URL: https://ieeexplore.ieee.org/document/10982630

[29] I.Kalinina, P. Bidyuk, A. Gozhyj, V. Gozhyi, V. Nechakhin, Approach to identification of anomalous values in analysis tasks and data pre-processing, in: Babichev, S., Lytvynenko, V. (eds) Lecture Notes in Data Engineering, Computational Intelligence, and Decision-Making, Volume 2. ISDMCI 2024. Lecture Notes on Data Engineering and Communications Technologies, vol 244. (2025). Springer, Cham. doi: 10.1007/978-3-031-88483-2_6.

[30] I. Kalinina, A. Gozhyj, P. Bidyuk, V. Gozhyi, M. Korobchynskyi, V. Nadraga, A systematic approach to data normalization and standardization in machine learning problems, in: Babichev, S., Lytvynenko, V. (eds) Lecture Notes in Data Engineering, Computational Intelligence, and Decision-Making, Volume 2. ISDMCI 2024. Lecture Notes on Data Engineering and Communications Technologies, (2025), vol 244. Springer, Cham. https://doi.org/10.1007/978-3-031-88483-2_11.

[31] I. Kalinina, A. Gozhyj, P. Bidyuk, V. Gozhyj, Multilevel ensemble approach in classification problems, conference: 2024 IEEE 19th International Conference on Computer Science and Information Technologies (CSIT). Lviv, Ukraine (2024) doi: 10.1109/CSIT65290.2024.10982625, URL: https://ieeexplore.ieee.org/document/10982625.

[32] P. Bidyuk , I. Kalinina, O. Zhebko, A. Gozhyj, T. Hannichenko, Classification system based on ensemble methods for solving machine learning tasks, CEUR- WS. (2023), vol. 3426. Pp. 1-11. CEUR-WS.org/Vol-3426/paper5.pdf. (ISSN 1613-0073).