Jerzy TCHÓRZEWSKI[1],
ORCID: 0000-0003-2198-7185

Tomasz MIELCARZ[2]
ORCID: 0009-0008-7290-8575

[1] Siedlce University of Natural Sciences and Humanities
Faculty of Exact and Natural Sciences
Institute of Computer Science
ul. 3 Maja 54, 08-110 Siedlce, Poland

[2] Student at University of Natural Sciences and Humanities
Faculty of Exact and Natural Sciences
Institute of Computer Science
ul. 3 Maja 54, 08-110 Siedlce, Poland

Selection of an algorithm for classifying data quoted on the Day Ahead Market of TGE S.A. in MATLAB and Simulink using Deep Learning Toolbox

DOI: 10.34739/si.2023.28.05

Abstract. The article contains an analysis leading to the selection of an algorithm for classifying data listed on the Day-Ahead Market of TGE S.A. in MATLAB and Simulink using Deep Learning Toolbox. In this regard, an introduction to deep learning methods, classification methods, and classification algorithms is provided first. Particular attention was paid to the essence of three important deep learning methods in the classification, i.e. the methods called: Stochastic Gradient Descent Momentum, Root Mean Square Prop and Adaptive Moment Estimation. Then, three architectures of artificial neural networks used in deep learning were characterized, i.e.: Deep Belief Network, Convolutional Neural Network and Recurrent Neural Network. Attention was paid to the selection parameters of algorithms for learning deep artificial neural networks that can be used in classification, such as: accuracy, information losses and learning time. Practical aspects of research experiments were also shown, including selected results of research conducted on volume and fixing 1 data quoted on the TGE S.A. Day-Ahead Market. After analyzing the obtained test results for the hourly system, it was noted that the least suitable algorithm for classification purposes was the Stochastic Gradient Descent Momentum algorithm, which in each case had worse results than the other two algorithms, i.e. the Adaptive Moment Estimation algorithm and the Root Mean algorithm Square Prop. However, the best algorithm turned out to be the Adaptive Moment Estimation algorithm, which obtained the highest accuracy, which was at a level comparable to the Root Mean Square Prop algorithm, with the latter algorithm having larger losses.

Keywords. Artificial Neural Networks, Data Classification, Day-Ahead Market, MATLAB and Simulink Environment, Selection of the classification algorithm.

1. Introduction

Due to the development of artificial intelligence methods, especially deep artificial neural networks and cluster analysis methods, it has become possible to analyze large data sets in order to search for regularities (or irregularities) in them [7, 11-12, 18, 21, 23 , 36, 38]. In deep learning methods, various architectures of artificial neural networks and various algorithms for learning artificial neural networks are used [2, 4, 17, 26, 37], and from the second point of view, machine learning methods of systems are used [1, 14, 24, 34]. On the one hand, the so-called large systems such as economic companies, state administration offices, interconnected social groups, etc. collect large data sets in order to analyze them for the purposes of managing economic corporations, governing the state, marketing or social activities, etc. [1, 5-6, 18, 22, 32, 36, 38].

The data collected by large systems sometimes reach huge sizes, reaching not only the size of gigabytes, terabytes or petabytes, but sometimes much larger sizes, and what's more, this limit is constantly growing. On the other hand, there is a noticeable rapid development of information and telecommunications technologies, which makes access to high-speed Internet connections and data processing tools more and more common and expected. Such a friendly ground is conducive to the dynamic development of methods for analyzing large data sets called Big Data. The processing of huge amounts of data and the richness of their content is not fully possible by a man who is not equipped only with IT and ICT technologies, which means that new automated methods are being developed and are being sought to enable their deep analysis and search for regularities and thus irregularities.

These methods currently include methods using machine learning on the one hand and deep artificial neural networks on the other [4, 14, 17-18, 24, 26, 37]. There are various limitations in this regard, including: the need to have more and more computing power, which, fortunately, only slightly hinders the analysis, because processors and graphics cards are developing equally quickly, and new methods, including new artificial intelligence methods, and even quantum-inspired artificial intelligence methods are developing just as fast intelligence and methods of quantum computing [30].

Large datasets include more and more often data quoted as part of transactions made on various types of stock exchanges, including e.g. on commodity exchanges, such as Towarowa Giełda Energii S.A. (TGE S.A.), which contains regularities that are very important for the purposes of the functioning and forecasting of the economy, including regularities of development in a certain scope of electricity generation, transmission and reception (EE). For these reasons, among others, data listed on the Day-Ahead Market (DAM) of TGE S.A. were selected for the study. and MATLAB and Simulink environment with Deep Learning Toolbox [2, 19, 34].

Thus, when the problem is already defined and the data structure and the purpose of its use are known, the problem of choosing a classification algorithm becomes an important research problem. Different approaches are used here, one is to assume that all constraints are bypassed and then test all available algorithms in different possible configurations using different prediction assumptions and then compare the results obtained. Although such an approach is substantively sensible, it is rarely used in practice, especially due to the time needed for such an extensive analysis. For these reasons, other methods related to the selection of an algorithm for classification are used in practice. the mathematical nature of the algorithm, the number of observations, the number of variables, the number of labels (categorical values) used, the required accuracy of the algorithm, learning time and constraints (e.g. model learning and updating time, model execution time after training, etc.).

Due to the fact that mathematical models of the system were obtained from the practice of the existing identification modeling of the Day-Ahead Market system, and the number of data listed on TGE S.A. on an hourly basis, it covers a relatively long period and the number of variables included in the analysis is not too numerous (volume of electricity and average volume-weighted electricity price obtained in 24 hours of a day), therefore it was decided to adopt three parameters in the analysis, i.e.: learning rate parameter ANN with the appropriate algorithm, learning accuracy parameter and MSE error [15]. However, three methods of teaching deep artificial neural networks were selected for experimental research, namely [2, 4, 17, 26, 37]:

- Stochastic Gradient Descent Momentum (SGDM), which is based on an iterative method for optimizing stochastic functions with the appropriate smoothness property,

- Root Mean Square Prop (RMSProp), which is a method that uses the root mean square property, this method is based on a gradient similar to SGDM, but here using an exponential mean of the gradient.
- Adaptive Moment Estimation (ADAM), which consists in estimating the adaptive moment.

2. Architecture and methods of learning deep artificial neural networks

Initially, efforts were made to limit the ANN architecture to as few hidden layers as possible, and in practice, one or at most two hidden layers were used to increase its capacity. However, over time, the idea of using more than two hidden layers returned, and attempts were made to change the configuration of hidden layers. Various other treatments were also undertaken, such as increasing the number of neurons in layers, or increasing the number of layers, and changing connections between neurons in layers and between layers, etc. [10, 20, 27-29]. This procedure was dictated by the requirements resulting from ever new types of input data and new expectations regarding the output data. This was followed by new types of layers in such deep artificial neural networks as:

1) Deep Belief Network (DBN), which uses the Restricted Boltzmann Machine (RBM) to create the so-called pre-trained layers of neurons,
2) Convolutional Neural Network (CNN), which uses new types of activation functions and changes the connections between layers of neurons,
3) Recurrent Neural Network (RNN), which uses connections that better model the time domain in the data series, etc.

In this way, hybrid architectures of artificial neural networks gradually appeared, and these inspired researchers to build deep networks based on various types of hybrid building blocks, such as: unidirectional multi-layer artificial neural networks, RBM machines that model the probability of a model leading to the separation of features (data moves in one direction) and autocoders, i.e. unidirectional neural networks in which additional loads are determined to calculate the reconstruction error of the original input data [4-5, 25, 30, 37, 39]. It is also worth mentioning two previously known but improved artificial neural networks, i.e.:

1) ANN created on the basis of the RNN network under the name of Long Short-Term Memory (LSTM) as a type of repetitive artificial neural networks, the advantage of which is to remember already processed data (the so-called problem with disappearing gradients has been eliminated in these networks), while LSTM networks have three gates: input gate, forget gate and exit gate, whose task is to decide about information

worthy of attention, such as the entrance gate decides what important information can be added from the current step, the forget gate decides which data should be omitted, and the gate decides the output finalizes operations [4, 15, 25-26].

2) Bidirectional Long Short-Term Memory (BiLSTM) as neural networks similar to LSTM, consisting of two LSTM networks: one taking input data in the forward direction and the other - in the reverse direction, with both SNNs connected to each other by the same output layer. Such a connection means that for each point in a given sequence, the artificial neural network of the BiLSTM type has full, sequential information about all points of the signal flow, before and after each neuron. Therefore, artificial neural networks of the BiLSTM type effectively increase the amount of information available in the ANN, improving the context available for the algorithm [15].

The goal of the Convolutional Neural Network (ConvNet, CNN) is to learn the higher-order characteristics of data using convolution[2], which is mainly used for two-dimensional data, although three-dimensional data is also sometimes studied. The CNN network is well suited for text analysis or image analysis thanks to the optical recognition of characters or whole words, or images, and even sounds. The image recognition efficiency is very high compared to other algorithms [4, 15]. Convolutional CNNs consist of many layers. After entering the input data, the so-called activation maps. They contain important information about the elements in the input. For example, if the input data are images, each neuron takes a fragment of pixels and then multiplies the value of their colors by its weights. They are successively summed up and passed on to the activation function layer. The first (bottom) layer of the convolutional Artificial Neural Network can detect features such as: width, height and vertical, horizontal and diagonal edges. More sophisticated CNNs are able to detect higher-level features, such as facial recognition [4].

[2] Convolution in mathematics, statistics, electrical engineering, signal theory, etc. is also called convolution (integral convolution, convolutional multiplication) similar to cross-correlation - it is the result of operation (convolution product, product) specified for two functions or signals described by the functions in the form of a result another function, which may in particular be a modified version of the original functions. This name is also used to describe the result of this operation, which is also called the convolution product (or product). There is also an operation inverse to convolution called deconvolution.

3. Deep learning algorithms for artificial neural networks

There are many algorithms that are used in deep learning, including its optimization. A properly selected algorithm can shorten the results by minutes, hours or even days, depending on the data, and also allows you to obtain appropriate test results.

Such an algorithm is the Stochastic Gradient Descent Momentum (SGDM), which is based on an iterative method for optimizing stochastic functions with the appropriate smoothness property. The operation of this algorithm is reduced to calculating a number of formulas by means of which information and even knowledge about the operation of the SGDM method used to learn an artificial neural network can be obtained. In order to obtain the appropriate relationships, the momentum, which is the moving average of the gradients, is first defined, with the use of two formulas described in detail in [15].

Another method of learning optimization is the Root Mean Square Prop (RMSProp) method. This method is based on a gradient similar to SGD with momentum, i.e. the concept of an exponentially averaged gradient is used here as a descending gradient with momentum. The difference is the need to update the parameters [19]. To describe the principle of operation of the RMSProp method, it is worth noting, among others, that there is an improved RProp algorithm here in such a way that mini-parts can be taught regardless of the problem of different size gradients.

The ADAM method of estimating the adaptive moment[3] was based on two algorithms: AdaGrad and RMSProp, which extends and performs its tasks much better than them [17]. An example is the RMSProp method, which only adjusts the learning rates of the parameters based on the first moment average as opposed to the ADAM method. Although this method also uses the average of the second moments of the gradients (off-centric variance), the parameters are then updated based on learning coefficients [19].

ADAM is a stochastic objective function optimization algorithm that is based on lower-order adaptive moment estimates. This method can be used to train artificial neural networks and deep learning [4]. Its big advantage may also be that the updated parameters are invariant for the scaled gradient, whose sizes are approximately limited by the step size hyperparameter. The ADAM method is a very efficient method of teaching artificial neural networks when the data sets are very large and contain many [15, 19].

[3]There are also other similar methods, including: NADAM and ADAMAX, which extend the ADAM method.

4. Data classification types

Data classification is the process of categorizing a given set of data into classes, which involves the elimination of multiple duplication of data in order to reduce storage and backup costs while speeding up the search process. Classification can be performed on both structured and unstructured data. The process begins with determining the class of the data points. These classes are often referred to as targets, labels, or categories. Data classification is of particular importance when it comes to managing risk, compliance and data security. The effectiveness of classification depends to a large extent on the type of input data, model construction or, above all, the use of an appropriate algorithm. There are two types of classification. The first is called type classification:

- Lazy Learners (LL), when learning data is stored until test data is available. In this case, classification is done using the most related data in the stored training data. They have more time to anticipate compared to willing students. The main algorithms using this type of classification is the k-nearest neighbors method [7, 11, 29, 38].
- Eager Learners (EL), in which the classification model is built on the basis of input data and data used in learning. Such a classification must be able to accept a single hypothesis that will work for the entire solution space, which takes longer to train. Algorithms using this model include: decision tree, naive Bayesian classifier, as well as Artificial Neural Networks [38].

The disadvantage of using artificial neural networks in classification, compared to other methods used, e.g. machine learning, is their more difficult and less unambiguous interpretation, as well as a very long learning time [9, 18, 21, 38]. However, the use of artificial neural networks has many advantages, such as the fact that they are:

- self-adaptive data-driven methods because they can adapt to data without explicitly specifying a functional or distributional form for the underlying model,
- universal functional approximations, because they can approximate any function with any accuracy,
- non-linear models, which makes them flexible in modeling complex relationships in the real world [39].

5. Classification evaluation measures

The classification results in the division of the data into classes which, according to the statisticians, correspond to labels for different populations, with class membership being determined independently of any specific attributes or variables. One can also note the definition of a class combined with the problem of prediction, and then the class is the result to be predicted based on the knowledge of the attributes, with the class being statistically a random variable. Classes are usually predefined by the division of the sample space, i.e. the attributes themselves, and in this sense a class is a function of the attributes. Thus, a crafted item may be classified as defective if some attributes fall outside predetermined limits [14].

Classification models are subject to evaluation. One of the basic evaluation measures is the so-called accuracy defined as the ratio of the number of correct predictions to the number of all forecasts, which can be called the prediction accuracy, i.e. to what extent the model is right. The predicted and actual classification can be represented in a table called a confusion matrix. It helps to visualize if the model is "confused" in distinguishing between two classes. There are two row and column labels.

They are called positive and negative. An example would be a 2 x 2 matrix, where the row labels represent the primary truth labels, while the column labels represent the predicted labels (how the matrix is represented exactly this way is not a rule). Each item is labeled with two words: true or false, positive or negative. True is when there is a match between the prediction labels and the underlying truth.

Falsehood is when there is a discrepancy between the labels of the prediction and the underlying truth. So whenever a prediction as positive is wrong, the labels true positive and false negative are used. Otherwise, it will be true negative and false positive. Knowing the definition of true positivity and negativity labels, you can apply them to the accuracy formula:

$$accuracy = \frac{true \quad positives}{true \quad positives + false \quad positives}. \tag{1}$$

It is worth adding that in the literature on the subject there are many types of classification, including the most frequently used pattern and patternless classification called cluster analysis or unsupervised learning [36], and, to generalize, pattern classification involves assigning objects based on selected features to previously defined n classes, and patternless classification (cluster analysis) consists in recognizing the structure of a set of objects based on data analysis by separating similar clusters of these objects on the basis of selected features (class identification), where the number of these clusters is not known a priori [38].

6. Loss of information

The loss function determines how far the model deviates from the predicted result. The task of the loss function is to reduce the number of misclassifications. The loss function works on error to quantify how bad it is to get an error of a certain size that is affected by negative consequences resulting in an incorrect prediction. There are various functions for this. His choice is very important. One of the loss-reducing functions is Cross-Entropy Loss, which measures the efficiency of the classification model. It outputs a number between <0, 1> to measure the difference between two probability distributions for the data. If the predicted probability differs from the actual label then it can be said that there are losses. Losses decrease as probability equals 1. Cross-Entropy Loss is determined as follows [15]:

$$L_{\mathrm{CE}} = -\sum_{i=1}^{n} t_i \log(p_i), \tag{2}$$

where:

n – number of classes,
t_i – truth label most often taken as the probability of a true class,
p_i – probability Softmax for i^{th} class,
i – i-th class index.

Softmax is a form of function that normalizes the input value to a vector of values consistent with the probability distribution, the sum of which is 1. The output values mean the probability in the interval <0.1>, similarly to cross-entropy losses [45]. The softmax function is used as an activation function in the output layer of neural network models that predict a polynomial probability distribution. This means that softmax is used as an activation function for multiclass classification problems [46]. The formula for the Softmax activation function is [15]:

$$f(i) = \frac{e^{Ai}}{\sum_{n=i}^{c} e^{An}} \tag{3}$$

where:

n – number of classes,
A^i ... A^n – linear results,
i – i-th current loss,

where the linear results of this exponential function are, in a sense, normalized so that the sum of the results equals 1. Thanks to the applied normalization, the output values can be interpreted as estimates of the probabilities of a given input signal belonging to particular classes [15, 38].

7. Research preparation

7.1. Description of the terms used

Before data analysis research is performed, it is worth showing the way of describing the experiment, including the legend used for the research results presented in the following figures (Figure 1).

Figure 1. Legend to the table showing the ratio of accuracy to iterations. Source: own study [2, 15].

Thus, in Fig. 1, the blue line indicates the smoothed learning accuracy. Due to data normalization, it is easier to show learning trends. Next, the blue line shows the accuracy of training without smoothing. A certain unfinished solution is the overlapping of both of the above. lines, which means that with a large number of iterations, they may overlap, and thus both lines may not be visible, with the smoothed accuracy course being the most common line. The third line, which is a dashed line with dots every 50 iterations, shows the accuracy of the classification in the entire validation set. Analogously to the line indicating the degree of accuracy, there is also a loss line depending on the number of iterations, the scale of which starts from 3.5 and ends at 0. The notations regarding losses are shown in Figure 2.

Figure 2. Loss to iteration table legend. Source: own study [2, 15].

Thus, in the figures with the results of data analysis, the orange line denotes smoothed losses[4], the salmon line - unsmoothed losses, and the dashed line with dots - the accuracy of the

[4] Smoothing is used to reduce the variance in the course of signals, e.g. time series or regression models, e.g. exponential smoothing is a method of processing a time series by reducing its variance using a weighted moving average of past values, with weights decreasing exponentially with the distance in time.

classification in the validation set. It is also worth noting that depending on the type of classification or regression network, errors will use RMSE or not.

To the right of the graphs are information such as (from top): Validation Accuracy Percentage and whether the training is complete. Later, you can notice information such as the start of learning and its execution time. The number of epochs corresponding to the entire data run, the number of iterations and the number of iterations multiplied by epochs are described in the following.

Another piece of information is the number of iterations at which the validation training data will be analyzed. The final information is: the type of resource used (GPU or CPU), the option to lower the learning rate, or the learning rate (Figure 3).

Results	
Validation accuracy:	97.90%
Training finished:	Reached final iteration
Training Time	
Start time:	14-Aug-2020 17:49:02
Elapsed time:	26 sec
Training Cycle	
Epoch:	8 of 8
Iteration:	248 of 248
Iterations per epoch:	31
Maximum iterations:	248
Validation	
Frequency:	30 iterations
Other Information	
Hardware resource:	Single CPU
Learning rate schedule:	Constant
Learning rate:	0.01

Figure 3. Sample information on the obtained results of data analysis. Source: own study [2, 15].

If the "Verbose" option is set to "true" in the learning options, you can also get additional information about the learning process depending on the type of artificial neural network. This information appears in a special GUI window with commands. For the classification network there are information such as: number of epochs, number of iterations, learning time, classification accuracy for mini-series, classification accuracy for validation data, loss in cross-entropy mini-series for multiclass classification problems with mutually exclusive classes, cross-entropy validation data for multiclass classification problems with mutually exclusive classes, basic learning factor. In the case of the regression network, the data will be similar, but instead of mini-series accuracy and validation, there will be RMSE error data.

7.2. Data used in the research

The data used in the research were quoted on the Day Ahead Market (DAM) of the Polish Power Exchange. (TGE S.A.)[5], and only hourly data on volume-weighted average electricity prices and the volume of electricity consumed and sold for a bilateral transaction from January 1, 2016 to November 12, 2020 (1,772 days, and each 24 hours a day, i.e. a total of 42,528 hours) [16, 35]. The data quoted on the DAM include columns with the following data: trading date, delivery date, delivery time and fixing price 1, the value of which is quoted in hours in [MWh]. The downloaded data was not complete, because in several cases there was no data for the second hour and in such situations they were supplemented as average values from adjacent hours.

7.3. Description of the deep learning model

Five layers shown in Listing 1 [2, 4] were used in deep learning. The first layer of "sequenceInputLayer", i.e. the input layer of the sequence determines how many inputs are entered into the Artificial Neural Network.

```
layers = [ ...
    sequenceInputLayer(1)
    bilstmLayer(100,'OutputMode','last')
    fullyConnectedLayer(1772)
    softmaxLayer
    classificationLayer]
```

Listing 1. Listing for a deep learning model. Source: own study [2, 15].

The second layer of "bilstmLayer" is a bidirectional Long short-term memory layer and is designed to learn from ANN dependencies on single sequential data or even entire data sequences. The bi-directionality for LSTM is that the learning ANN is fed the original data end to end and then end to end. The values that can be set in the "bilstmLayer" layer is the number of hidden units, which corresponds to the amount of information stored between time steps, including information from all previous steps, regardless of the length of the sequence. The next value determines the name of the layer, and the last one determines the format of the output

[5] This is a company that provides the ability to trade transactions for mass volumes of sales and purchases of electricity or certificates. This company has been certified to carry out these activities since 2003. In addition to energy trading, TGE also offers the possibility of exchanging liquid fuel and gas goods.

data, where you can choose: "sequence" when entering the entire sequence or "last" when entering the last step of the sequence.

The third layer is the layer: "fullyConnectedLayer", whose task is to multiply the input data by the matrix of weights, and then add the bias vector. You can specify the output size as the value of this layer. Depending on the tests carried out for hours or days, this value will change. In the first case, it will be 24, and in the second, 1 772, etc.

The fourth, penultimate layer is the layer: "softmaxLayer", whose task is to impose the Softmax function on the input. This function turns a vector of reals K into a vector of reals K summing to 1 regardless of whether the input is positive or negative, because they are always converted to a number between 0 and 1 to be interpreted as probabilities. Another important task of the Softmax function is to convert the probability distribution in such a way that it can be displayed by the user or passed on.

Finally, the fifth and last layer is the layer: "classificationLayer" which determines the cross-entropy loss for classification and weighted classification tasks with mutually exclusive classes. To classify a class from the output size of the previous class, include a combined layer of output size K and a Softmax layer before the "classificationLayer" layer.

8. Selected research results

Research was carried out for the volume of supplied and sold electricity [MWh] and the volume-weighted average prices of electricity according to the prices of fixing 1, fixing 2 and for continuous quotations, with this article only the results concerning fixing 1. The obtained results were divided into appropriate sets in order to use them in the appropriate learning of the ANN broken down into individual days and hours, with the assumed size of 64 mini-series and 900 [15].

8.1. Selected results for individual hours of the day

First, the results of deep learning for the following algorithms will be presented: SGDM, RMSProp and Adam in the case when data such as: volume and price of fixing 1 for individual hours of the day were entered into the SSN input. For hourly training, 900 epochs were used and the default mini-series size was adjusted to 24 data series (hours). Therefore, using the formula for the total number of iterations (2), it can be concluded that its number is 900, so the selection is correct.

8.2. Selected results for the fixing rate (price) 1

The results of the data analysis presented in this article concern the fixing price 1 (volume weighted average price ee)[6] for the Stochastic Gradient Descent Momentum (SGDM) learning algorithm. The learning process lasted 1 hour, 5 minutes and 28 seconds, and the accuracy of the mini-series at the end of the learning process was 50%, with a loss of 2.1594.

The course of hourly learning accuracy depending on the number of iterations is shown in Figure 4, which is characterized by variable accuracy in relation to the number of iterations, with repeated gradual increases and decreases in accuracy, respectively for the number of iterations: an increase to 140 iterations, and then a decrease to 200 iterations to again after 14 ups and downs reach a final accuracy of 40 for 900 iterations.

It can also be seen that the accuracy from the 600th iteration with some fluctuations remained relatively even. In summary, an exponential increase in average accuracy can be noted, characterized by para-periodic variations. The next graph in Figure 5 shows an error (loss) waveform that decreases to the 400th iteration with a slight increase upwards, with successive drop-off disturbances occurring at the 540th iteration, 670th iteration and 800th iteration.

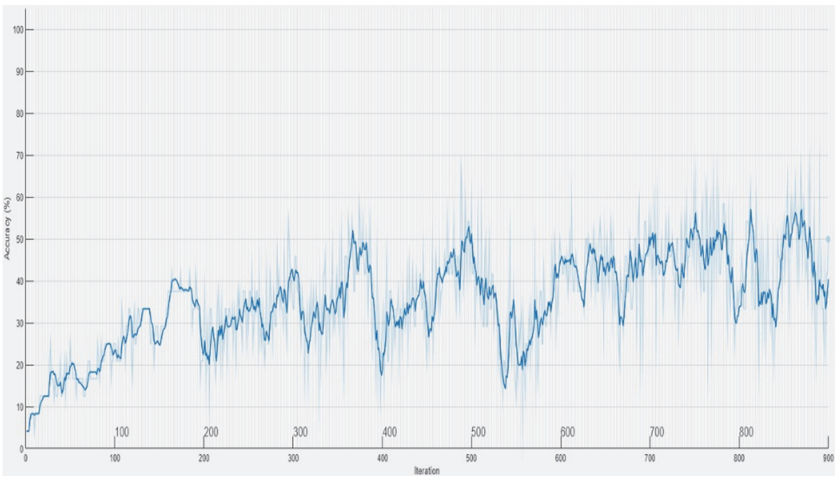

Figure 4. Accuracy graph of the hourly learning of the Fixing 1 course using the SGDM method. Y axis - accuracy, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

---

[6] Fixing I, i.e. the moment when energy prices are determined discreetly during individual hours of the day. This takes place every day at 10:30. Each Day Ahead Market participant submits orders to buy or sell electricity for specific hours of the day. In addition to Fixing I quotations, there are also Fixing II quotations and continuous quotations.

The second method used was the Root Mean Square Prop (RMSProp) method. The entire learning process for this algorithm took 51 minutes and 18 seconds, the final accuracy is 83.33%, and the losses are 0.4590.

Cyclic increases and decreases in accuracy were noted, with the average accuracy increasing exponentially to about 85% (Figure 6), and for the number of iterations above 480, there were times when the accuracy was 100%. On the other hand, the course of average loss values depending on the number of iterations shown in Figure 7 decreased to the average value of 0.5, which it reached for 900 iterations.

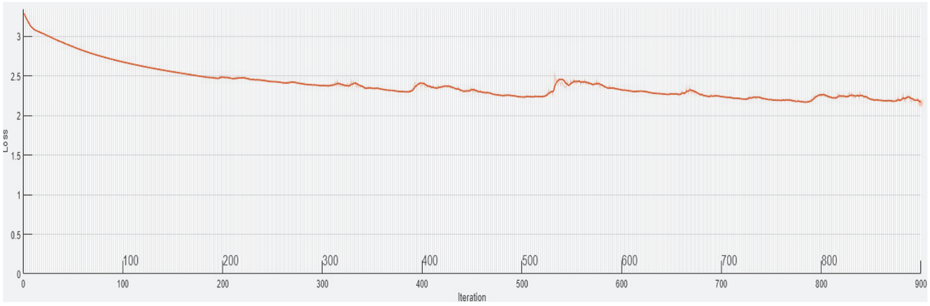

Figure 5. Hourly error (loss) chart obtained for fixing rate 1 (volume-weighted average ee price) obtained using the SGDM method. Markings: Y axis - losses, X axis (top scale) number of epochs (max 900), X axis (bottom scale) number of iterations (max 900). Source: own study [2, 15].

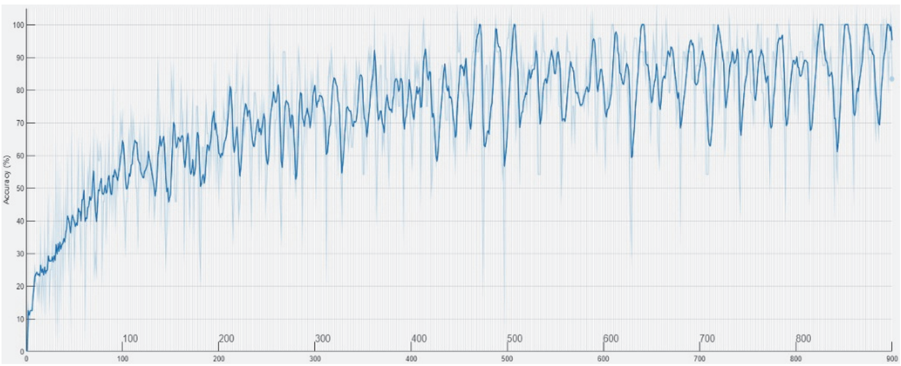

Figure 6. Fixing 1 course teaching accuracy chart using the RMSProp method. Y axis - accuracy, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

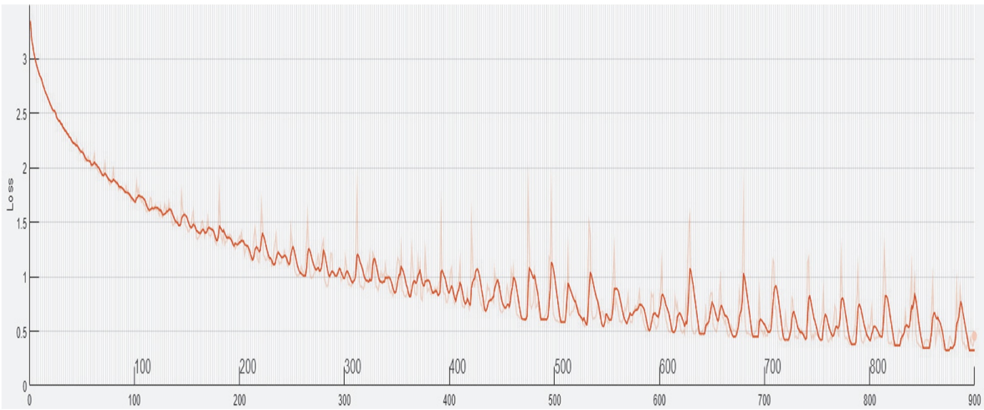

Figure 7. Chart of learning losses per hour for the Fixing 1 course using the RMSProp method. Y axis - losses, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

The third and last learning method used in the data analysis process was the Adam method. This algorithm learned for 59 minutes and 54 seconds, and its final mini-series accuracy score was 87.50% (Figure 8) with an average loss of 0.4646 (Figure 9). A certain similarity can be noticed in the course of the accuracy curve as well as the loss curve to the waveforms corresponding to the RMSProp method, for which the accuracy, despite fluctuations, was constantly increasing, and the losses decreased from the 150th iteration to 900 iterations with increasing changes (decrease by an order of magnitude).

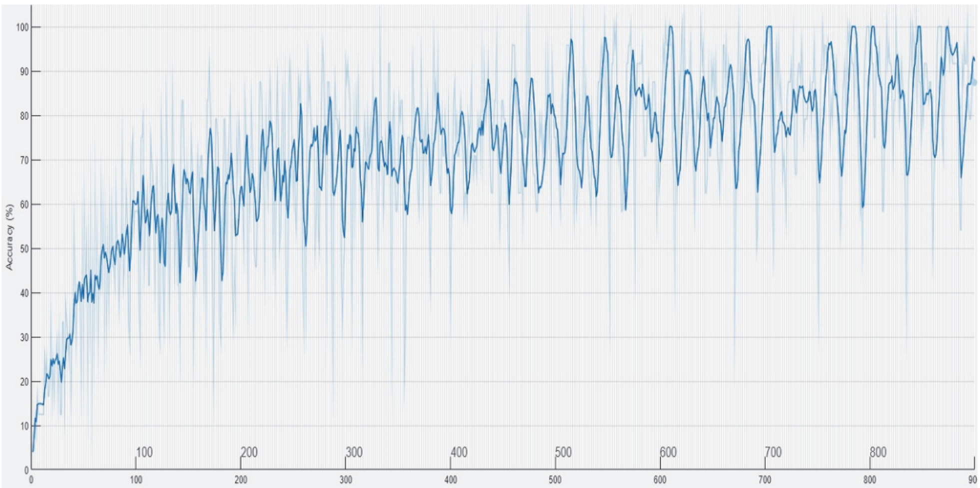

Figure 8. Graph of learning accuracy in the Fixing 1 course by the hour with the Adam method. Y axis - accuracy, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

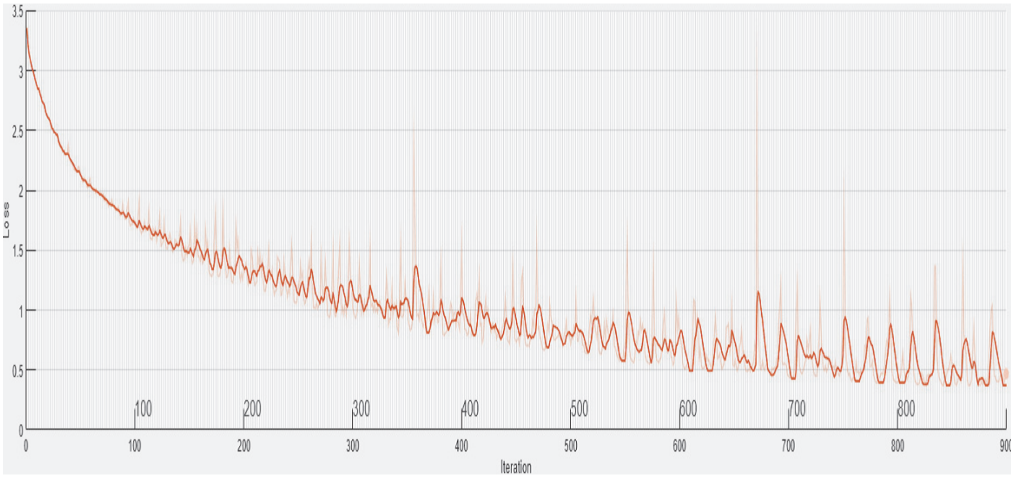

Figure 9. Graph of learning losses by hour for the Fixing 1 course using the Adam method. Y axis - losses, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

Summary of the obtained analysis results for the above-mentioned. three methods, i.e. for the SGDM, RMSProp and Adam methods, for data on the exchange rate (volume-weighted price) of electricity are presented in Table 1.

Table 1. Comparative list of learning results by hour for data on the fixing rate (price) 1. Source: own study [2, 15].

Algorithm name	Learning time	Accuracy	Losses
SGDM	01:05:28	50%	2.1594
RMSProp	00:51:18	83.33%	0.4590
Adam	00:59:54	87.5%	0.4646

The comparative analysis of the results obtained for the fixing 1 price (price) shows, among others, that the highest accuracy was achieved by artificial neural networks learned using the Adam (87.5%) and RMSProp (83.33%) algorithms, while for the RMSProp accuracy was lower than Adam algorithm, but id losses were lower Adam by 4.17% the former the former has worse accuracy by 16.67%, but its losses were lower by 0.0056.

8.3. Results obtained for fixing volume 1

The SGDM method for training for fixing volume 1 data did not work at all. Its operation time was 1 hour, 29 minutes and 25 seconds. The resulting final mini-series accuracy is 8.33% and the losses are 3.0066. The accuracy plot (Figure 10) shows slight deviations at the 30th, 150th, 250th and 860th iterations. In other places it remains the same and amounts to about 8.33%. The losses (Figure 11) tend to decrease to the 100th iteration and then remain at the same level, but they continue to decrease with a trend of several to ten thousandths of a loss unit, which cannot be seen with the naked eye. Low accuracy that had a low rise followed by a dip may be due to it not fitting well enough. This happens when the model is too simple and the data in the set is too complex.

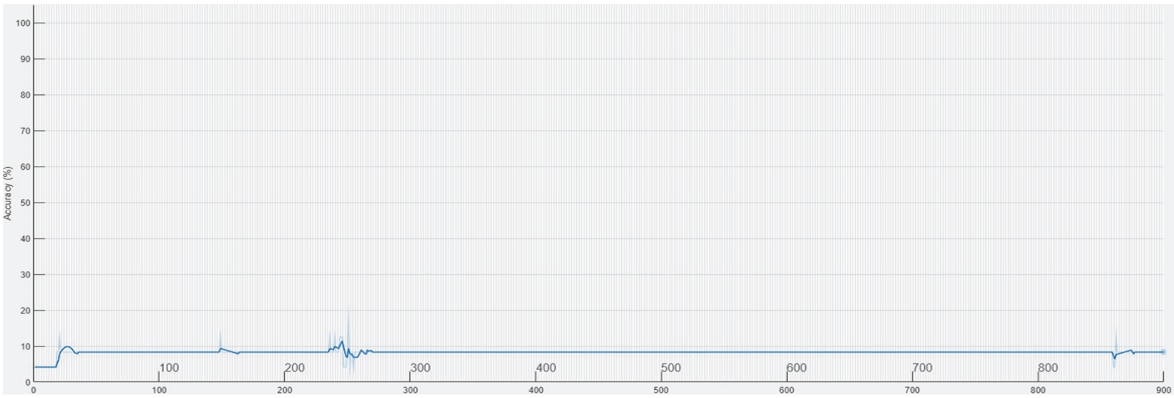

Figure 10. Chart of learning accuracy by the hour for the Fixing 1 volume with the SGDM method. The phenomenon of insufficient fit. Y axis - accuracy, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

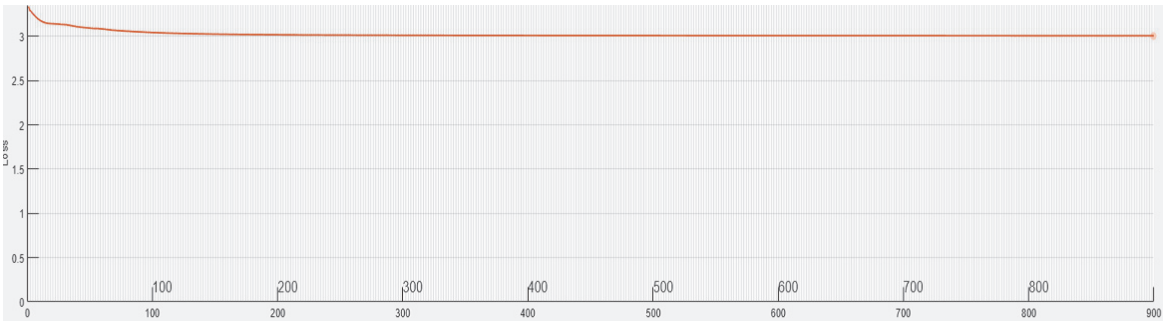

Figure 11. Chart of learning losses by the hour for the Fixing 1 volume using the SGDM method. Y axis - losses, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

The second method, RMSProp took 46 minutes and 12 seconds to learn. During this time, mini-series accuracy of 12.50% and losses of 2.6773 were recorded. The learning accuracy of fixing volume mini-series 1 (Figure 12) tends to increase despite fluctuations. The situation of losses is similar (Figure 13), where significant decreases occur in the first few iterations, and then fluctuations begin around the 100th. However, the losses are on the decline.

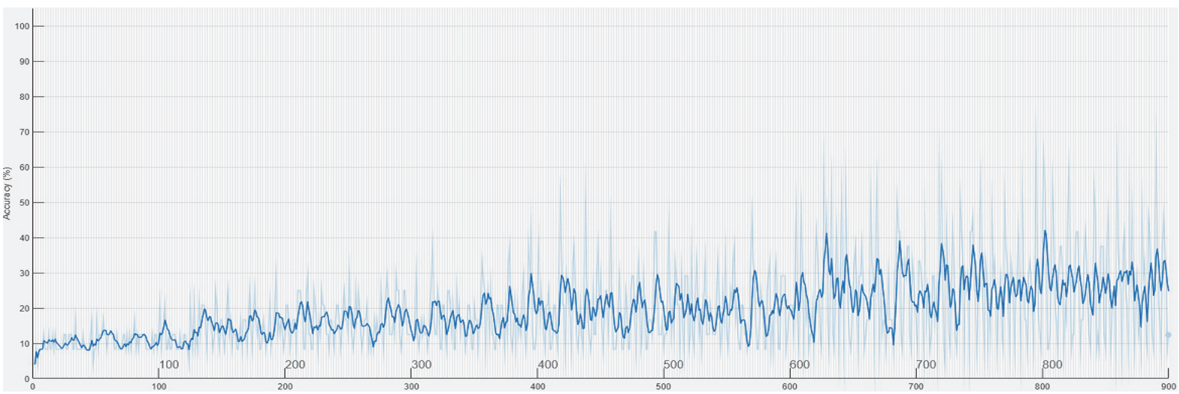

Figure 12. Hourly learning accuracy chart for Fixing 1 volume with RMSProp method. Y axis - accuracy, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

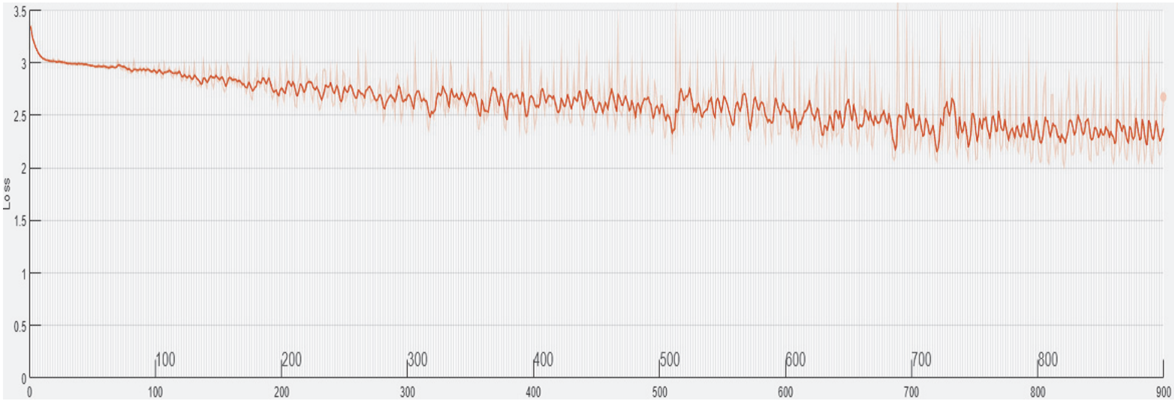

Figure 13. Chart of learning losses per hour for the Fixing 1 volume using the RMSProp method. Y axis - losses, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

The third and final learning method was the Adam algorithm, which achieved the best results for 1 hour, 8 minutes and 29 seconds, with a mini-series value of 75% and losses of 0.7796. The accuracy plot (Figure 14) tends to increase, but there are decreases at about the 230th, 320th, 530th, 770th and 840th iterations. Losses (Figure 15) also decrease, but there are several iterations where they increase, i.e. for the 40th, 70th, 140th, 320th, 430th, 520th, 760th and 870.

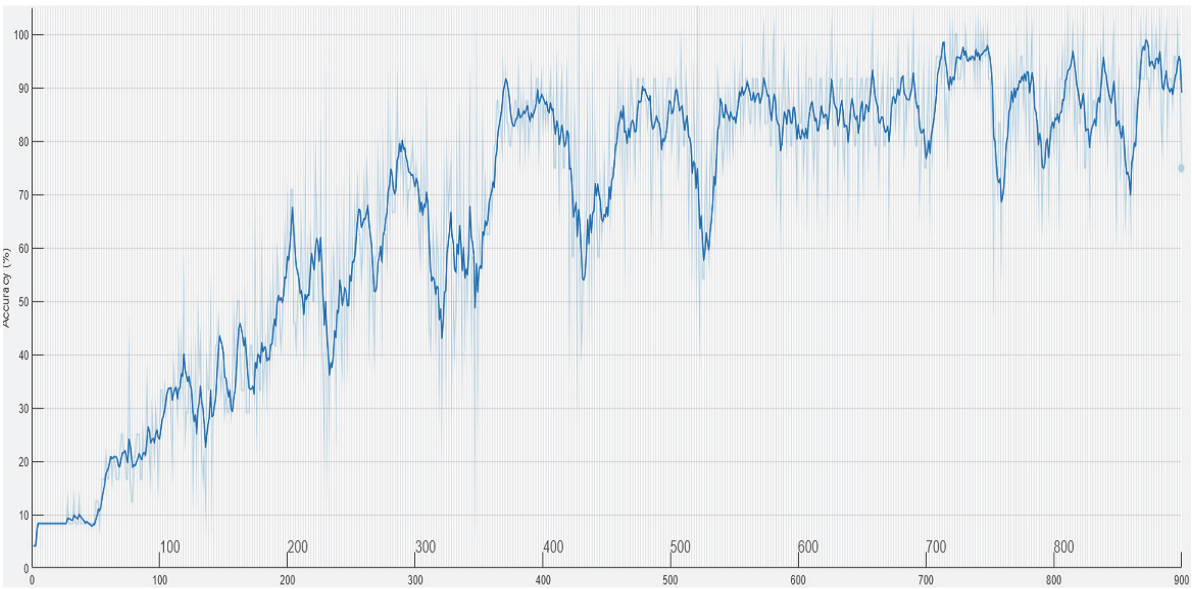

Figure 14. Accuracy chart of Fixing 1 volume hours learning with the Adam method. Y axis - accuracy, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

Figure 15. Fixing 1 volume hour learning loss chart using the Adam method. Y axis - losses, X axis at the top - number of epochs (max 900), X axis at the bottom - number of iterations (max 900). Source: own study [2, 15].

Summary of the obtained analysis results for the above-mentioned. three methods, i.e. for the SGDM, RMSProp and Adam methods, for data on the volume of electricity recorded for Fixing I are presented in Table 2.

Table 2. Table comparing the results of learning hours for fixing volume data 1. Source: own study [2, 15].

Algorithm name	Learning time	Accuracy	Losses
SGDM	01:29:25	8.33%	3.0066
RMSProp	00:46:12	12.5%	2.6773
Adam	01:08:29	75.0%	0.7796

From the comparative analysis of the analysis results obtained for the fixing volume 1, it can be seen, among others, that the highest accuracy was obtained by artificial neural networks trained using the Adam algorithm (75.0%), which obtained much higher accuracy than the RMSProp algorithm by as much as 63.0%, and definitely lower losses than the RMSProp algorithm - by as much as 1.8977 and from the SGDM algorithm - by as much as 2.2270.

9. Summary and directions of further research

In summary, it is worth noting that at the time of writing the article (Q1 2023), there were the following instruments[7]:

[7] [https://doradcapv.pl/rdn-rynek-dnia-nastepnego-energii-elektryczna-tge-ceny-dane-trendy/], while TGE S.A. plans to introduce a new trading schedule for electricity on the trading floor, which is to come into force on October 1, 2023, and will involve the abandonment of fixing I at 8:00 a.m., which will result in a reduction in the number of fixings from three to two. This means that the determination of the single price for the domestic market

1) Hourly instruments relating to individual hours (of 24 hours of the day) on the Polish market as instruments appropriate for Fixing I, when quotations take place from 8:00 to 10:30 one day before the delivery day, security is checked, quotations take place on single-price system every day with a price limit when using the X-Stream Trading IT system;

2) Hourly instruments regarding the Day Ahead Market in terms of the international market as an instrument appropriate for Fixing II - DAM, when quotations take place from 8:00 a.m. to 12:00 p.m. one day before the delivery day without checking the collateral, quotations take place in the single-price system according to PCR model (daily quotes - price limit applies - Sapri Trade IT system);

3) Block instruments such as: BASE (24 hours), PEAK (15 hours), OFFPEAK (9 hours), when quotations take place from 8:00 a.m. to 3:30 p.m. two days before the delivery day, collateral is checked, quotations are continuous daily and there is a price limit;

4) Weekend instruments type: BASE_WEEKEND; PEAK_WEEKEND; OFFP_ WEEKEND, when quotations take place from 8:00 a.m. to 3:30 p.m. two days before the delivery day, security is checked, quotations are continuous on Thursdays and Fridays and a price limit applies.

As part of the research carried out, the results of which are included in the article, regarding data quoted on the Day-Ahead Market for Fixing 1 in terms of the volume of electricity delivered and sold and the volume-weighted average price of electricity, the learning speed of deep artificial neural networks, their accuracy and losses resulting from learning process using three learning algorithms used in classification, i.e. for the following algorithms: SGDM, RMSProp and Adam.

For the purposes of the conducted research, relevant numerical data quoted on the Day-Ahead Market of TGE S.A. were obtained and prepared, i.e. in terms of the volume of electricity supplied and sold and the volume-weighted average price expressed at the fixing rate 1. Therefore, the obtained numerical data were first properly prepared for classification missing data were supplemented, and after their preparation, they were subjected to research experiments.

Conducting the research experiment involved designing the appropriate Deep ANN learned with the appropriate method, and then its implementation in the MATLAB and Simulink

---

will take place at one fixing, while the quotation schedule for market coupling with the NordPool Spot exchange remains on the existing principles [https://www.cire.pl/ artykuly/serwis-informacyjny-cire- 24/90563].

environment using the Deep Learning Toolbox. The data was processed in an appropriate way so that it could be classified, and then an artificial biLSTM neural network was designed. With its help, using the algorithms: SGDM, Adam, RMSProp, the relevant numerical data were tested, and the results obtained were compared in terms of the speed of learning artificial neural networks and their accuracy and losses for the volume and fixing rate 1.

After the analysis of the obtained test results for the hourly system, it was noted that the least suitable algorithm for classification purposes turned out to be the SGDM algorithm, which in each case had worse results than the other two algorithms, i.e. the Adam and RMSProp algorithms. On the other hand, the best algorithm for such data analysis turned out to be the Adam algorithm, which obtained the highest accuracy, but at the level of the RMSProp algorithm, with slightly greater losses.

In further research, the behavior of deep learning algorithms for fixing 2 and continuous hourly trading can be examined, similarly to the research presented in the article for fixing 1. In addition, the behavior of this type of deep learning algorithms for data not only quoted in on an hourly basis as well as on a daily basis. It is also worth conducting research on a larger scale by increasing the number of epochs, especially in the case of data for which the accuracy was much lower than 95%, or even 70-80%. If the results still did not meet expectations, it would be possible, for example, to carry out additional preliminary treatments on the data, e.g. by normalizing data and removing outliers.

A new direction of research could be data analysis using the computing power of graphics cards, clouds and clusters. The obtained results would allow to compare the speed of effective learning using different platforms for data concerning not only DAM TGE S.A., but e.g. for all electricity commodity markets operating in Europe. An important direction of the analysis could be the extension of deep learning with cluster analysis (clustering) and regression, and then comparing them with the classification, so that in the future it would be possible to find the best way to analyze data for data listed on the Day-Ahead Market of TGE S.A.

References

1. P. Cichosz: Systemy uczące się [Learning systems; in Polish], WNT, Warszawa 2000, pages 894.
2. H. B. Demuth, M. T. Hagan, M. H. Beale, Deep Learning Toolbox™ User's Guide" MathWorks III, 2020, pp. 112-337.
3. K. Domeradzki, A. Niewiadomski: Object classification with artificial neural network: A comparative analysis, Studia Informatica. Systems and Information Technology, No 1-2(23)2019, pp. 43-56.

4. A. Gibson, J. Patterson, A. Gibson: Deep Learning. Praktyczne wprowadzenie, Helion, Warszawa 2018, pages 451.
5. A. Graves and J. Schmidhuber: Framewise phoneme classification with bidirectional LSTM networks, Proceedings. 2005 IEEE International Joint Conference on Neural Networks, Vol. 4/2005, Montreal, QC, Canada, 2005, pp. 2047-2052.
6. M. Iwanowski: Rozwój algorytmów poprawiających skuteczność klasyfikacji w zastosowaniu do danych przestrzennych [Development of algorithms that improve the efficiency of classification when applied to spatial data; in Polish], Politechnika Warszawska, Warszawa 2015, pp. 14-15.
7. K. Jajuga, J. Nazarko [red. nauk.]: Klasyfikacja i analiza danych. Wybrane zagadnienia [Data classification and analysis. Selected Issues; in Polish], Rozprawy Naukowe nr 120, Wydawnictwo Politechniki Białostockiej, Białystok 2005, pages 141.
8. J. Kacprzyk, J. Korbicz, P. Kulczycki: Automatyka, robotyka i przetwarzanie informacji [Automation, robotics and information processing; in Polish], WN PWN, Warszawa 2020, pages 768.
9. M. Kłopotek: *K*-means is probabilistically poor, Studia Informatica. Systems and Information Technology, No. 2 (27)2022, pp. 5-26.
10. J. Korbicz, A. Obuchowicz, D. Uciński: Sztuczne sieci neuronowe. Podstawy i zastosowania [Artificial neural networks. Fundamentals and applications; in Polish], Problemy Współczesnej Nauki, Teoria i Zastosowania, Informatyka, Akademicka Oficyna Wydawnicza PLJ, Warszawa 1994, pages 251.
11. J. Koronacki, Ćwik J.: Statystyczne systemy uczące się [Statistical learning systems; in Polish]. WNT, Warszawa2005, pages 292.
12. D. T. Larose: Odkrywanie wiedzy z danych. Wprowadzenie do eksploracji danych [Discovering knowledge from data. Introduction to data mining; in Polish]. WN PWN, Warszawa 2006, pages 227.
13. MATLAB® Creating Graphical User Interfaces, MathWorks, 2015-2023.
14. D. Michie, D. J. Spiegelhalter, Ch. Taylor: Machine Learning, Neural and Statistical Classification, IEEE Transactions on Systems Man and Cybernetics, Part C Applications and Reviews, 1999, pages 299.
15. T. Mielcarz: Wykorzystanie środowiska MATLAB oraz Deep Learning Toolbox do klasyfikacji danych systemu Rynku Dnia Następnego TGE S.A. [The use of the MATLAB environment and the Deep Learning Toolbox to classify the data of the Day-Ahead Market system of TGE S.A; in Polish], praca magisterska napisana w Instytucie

Informatyki pod kierunkiem dr hab. inż. Jerzego Tchórzewskiego, prof. uczelni, Wydział Nauk Ścisłych i Przyrodniczych, UPH w Siedlcach, Siedlce 2022.

16. W. Mielczarski: Rynki energii elektrycznej. Wybrane aspekty techniczne i ekonomiczne [Electricity markets. Selected technical and economic aspects; in Polish], Agencja Rynku Energii S.A., Warszawa 2000, pages 321.
17. M. Mikołajczyk, A. Kwasigroch, M. Grochowski: Analiza sceny przy użyciu głębokich sieci neuronowych typu YOLO [Scene analysis using deep YOLO neural networks; in Polish], XXIX Seminarium Zastosowanie Komputerów w Nauce i Technice 2019", O/Gdański PTETiS, Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej, Nr 68/2019, pp. 37-40.
18. T. Morzy: Eksploracja danych. Metody i algorytmy [Data mining. Methods and algorithms; in Polish], WN PWN, Warszawa 2021, pages 564.
19. N. Papakostkas, C. Constantinescu, D. Mourtzis: Novel Industry 4.0 Technologies and Applications, Applied Sciences, 2020, pp. 77-79.
20. S. Osowski: Sieci neuronowe do przetwarzania informacji [Neural networks for information processing; in Polish], Wydawnictwa Naukowo-Techniczne, Warszawa 2013, pages 422.
21. S. Osowski: Metody i narzędzia eksploracji danych [Data mining methods and tools; in Polish], Wydawnictwo BTC, Legionowo 2013, pages 388.
22. M. Skorzybut, M. Krzyśko, i inni: Systemy uczące się. Rozpoznawanie wzorców, analiza skupień i redukcja wymiarowości [Learning systems. Pattern recognition, cluster analysis and dimensionality reduction; in Polish], WNT, Warszawa 2009, pages 400.
23. K. Stąpor: Automatyczna klasyfikacja obiektów [Automatic classification of objects; in Polish]. Problemy współczesnej nauki. Teoria i zastosowania. Informatyka, AOW EXIT, Warszawa 2005, pages 267.
24. P. Świtalski, M. Kopówka: Machine learning methods in e-mail spam classification, Studia Informatica. Systems and Information Technology, No 1-2(23)2019, pp. 57-76.
25. N. Qian: On the Momentum Term in Gradient Descent Learning Algorithms. Neural networks. The Official Journal of the International Neural Network Society, 1999, pp. 1-7.
26. J. Tabor, M Śmieja, Ł. Struski, p. Spurek, M. Wołczyk: Głębokie uczenie. Wprowadzenie [Deep learning. Introduction; in Polish], Helion, Gliwice 2022, pages 179.

27. R. Tadeusiewicz: Archipelag sztucznej inteligencji [Artificial intelligence archipelago; in Polish], AOW EXIT, Warszawa 2021, pages 126.
28. R. Tadeusiewicz: Sztuczne sieci neuronowe [Artificial neural networks; in Polish], AOW RW, Warszawa 1993, pages 195.
29. J. Tchórzewski, T. Kania: Comparative study of cluster analysis methods used to evaluate systems in the power industry, chapter 1 in: J. Tchórzewski, P, Switalski, Theory and Application of Artificial Intelligence Methods, Monograph No. 1 in series Intelligent Systems and Information Technologies, Wydawnictwo Naukowe UPH, Siedlce 2022, pp. 11-42.
30. J. Tchórzewski: Metody sztucznej inteligencji i informatyki kwantowej w ujęciu teorii sterowania i systemów [Methods of artificial intelligence and quantum computing in terms of control theory and systems; in Polish], Wydawnictwo Naukowe UPH, Siedlce, 2021, pages 343.
31. J. Tchórzewski, I. Kwiczak: Mapowanie informacji z baz danych za pomocą sieci neuronowych samoorganizujących się [Mapping information from databases using self-organizing neural networks; in Polish]. Studia Informatica. Systems and Information Technology. Volume 1(3)2004, pp. 99-105.
32. J. Tchórzewski, M. Kłopotek, M. Kujawiak: Studium porównawcze metod prowadzenia odkryć [A comparative study of discovery methods; in Polish]. Studia Informatica. Systems and Information Technology. Volume 2(4)2004, pp. 105-122.
33. J. Tchórzewski, M. A. Kłopotek: A case study in neural network evolution, Prace Naukowe IPI PAN, No. 943, pp. 1-12.
34. J. Tchórzewski, B. Longota: Analysis of data quoted on the Day-Ahead Market of TGE S.A. using Statistics and Machine Learning Toolbox, Studia Informatica. Systems and Information Technology, Volume 2(27)2022, pp. 49-74.
35. Towarowa Giełda Energii S.A., https://tge.pl/energia-elektryczna-rdn [accessed: 14.08.2023].
36. J. J. Trajer, M. Janaszek-Mańkowska, D. R. Mańkowski: Komputerowa analiza danych w badaniach naukowych [Computer data analysis in scientific research; in Polish], Wydawnictwo SGGW, Warszawa 2016, pages 131.
37. S. Weidman: Uczenie głębokie od zera. Podstawy implementacji w Pythonie [Deep learning from scratch. Python implementation basics; in Polish], Wyd. Helion, Warszawa 2020, pages 219.

38. S. T. Wierzchoń, M. A. Kłopotek: Algorithms of Cluster Analysis, Monograph Series No. 3, Institute of Computer Science Polish Academy of Sciences, Warsaw 2015, pages 308.

39. P. G. Zhang: Neural Networks for Classification. A Survey, IEEE, 2000, pp. 451-452.