WORLD OF HEALTH 2020; (3) 13-19

DATASET SIZE-BASED APPROACH IN DESIGN OF ARTIFICIAL NEURAL NETWORK FOR BREAST CANCER DIAGNOSIS

Ivan Lorencin[1]*, Nikola Anđelić[1], Sandi Baressi Šegota[1], Daniel Štifanić[1], Jelena Musulin[1], Vedran Mrzljak[1], Elitza Markova-Car[2], Zlatan Car[1]

1 University of Rijeka, Faculty of Engineering, Vukovarska 58, 51000 Rijeka, Croatia
2 University of Rijeka, Department of biotechnology, Radmile Matejčić 2, 51000 Rijeka, Croatia

**Autor za korespodenciju:*
Ivan Lorencin, ilorencin@riteh.hr
University of Rijeka Faculty of Engineering, Department of automation and electronic,Vukovarska 58, 51000 Rijeka, Croatia

ABSTRACT

One of the challenges in medical data classification is the determination of the sufficient training dataset size. In this research, the effect of training dataset size on various artificial neural network (ANN) configurations is shown. All ANNs are trained and tested using Wisconsin Breast Cancer (Diagnostic) Dataset, which contains 569 samples. Dataset is divided into training dataset of 410 samples and test dataset of 159 samples. The training procedure for all ANNs is performed with datasets in the range between 10 and 410 samples. Performances of all ANNs are evaluated using ROC analysis. From obtained results, it can be seen that if datasets smaller than 282 samples are used for training of ANN, higher AUC values will be achieved if deep ANN designed with ReLU activation function is used. On the other hand, if larger datasets are used, the best performances will be achieved if ANNs are designed with one hidden layer and Logistic sigmoid activation function. Our results showed, that it is possible to use smaller datasets for classifier training if right ANN architecture is utilized. Furthermore, it can be concluded that there is no need for using large datasets for designing an ANN for breast cancer diagnosis.

Keywords: Activation function; Artificial neural network; Breast cancer diagnosis; Dataset size

INTRODUCTION

Artificial intelligence (AI) is today established as a powerful tool for medical diagnosis. AI algorithms, in collaboration with medical professionals, are providing a modern approach in the medical diagnostic. These tools are fast and effective in producing efficient diagnostic methods used in various medical specialities, as well as other fields (1-3). One of the AI methods used in medicine is data classification. Data classification is based on algorithms called classifiers (4). Classifiers are algorithms which perform separation of data into different groups according to the characteristics of that data. One of the most popular classification algorithms used in medicine is artificial neural network (ANN) (5, 6). A training dataset is needed for every classifier. Training procedure involves fitting classifiers parameters according to input data and output data. Input data represent values by which classifications are performed. Output data contain binary representation of class in which sample is classified (5). It has been shown that such techniques can be used to a great effect in the field of medicine (6, 7). In recent years, the trend of using large datasets for classifier training can be occurred. In case of ANN for handwriting recognition, training dataset of 60,000 samples is used (8). For hard characters classification, a training dataset of 52,900 images is used (9) A training dataset of 30,880 images for plant diseases diagnostics by leaf image classification is used (10). In medicine, a training dataset of 13,584 samples is used for gastric cancer diagnostic by means of endoscopic images (11). In addition, a dataset of 8,428 samples is used for training a classifier for esophageal cancer diagnostic, which is based on the convolutional neural network (CNN) (12). A dataset of 11,112 samples is used for automatic fracture detection by using radiographic images (13). In marine biology, a dataset of 10,728 images and 1489 videos is used for marine organism classification (14). In the case of the classifier for apple tree disorders, a training dataset of 2,539 images is used (15). For purposes of reducing carbon emissions, an AI-based water consumption meter is designed. In this case, a training dataset of 200,000 samples is used (16). For fruit classification, a classifier trained with 1,653 images is used (17).

Although tendency of using large training datasets, some studies use training datasets with less than 1,000 samples. In study of detecting fruit defects using ANN dataset of 400 samples is used. 96.75% accuracy is achieved using this approach (18).

During training, two challenges from a dataset size standpoint can be noticed:

- The first challenge represents acquisition of a sufficiently large training dataset needed for classifier to perform with desirable accuracy.
- The second challenge can be represented as finding the optimal dataset size with the aim of minimizing training time with keeping classifier accuracy.

For these reasons, analysis of classifier performances according to training dataset size is performed.

For training dataset size in range between 1,000 and 243,000 shows how performance improvements become less noticeable when datasets larger than 27,000 samples are used. These characteristics are defined by using Tweet Sentiment Labeling dataset and can be noticed for various classifier designs (19). Study of dataset size effect on the ANN for myocardial infarction diagnostic has shown

that reduction of dataset size by 83% makes no effect on ANN performances. Besides, significant reduction in training time has occurred (20). For small dataset sizes, it can be expected that classification with ANN will achieve higher accuracy than discriminant analysis (21). Additionaly, in landslide susceptibility assessment, better classification performances are achieved if Naive Bayes classifier is used instead of Logistic Regression classifier. These results are consequence of training on small dataset (22).

In this study, The Wisconsin Breast Cancer (WBC) Dataset is used. If the deep believe neural network is used for classification of aforementioned dataset, 99.68% accuracy is achieved (23). If Support Vector Machine (SVM) is used, classification accuracy from 97.07 to 98.10% is achieved. By using Radial Based Function and Linear Logistic Regression, the accuracies of 96.49% and 96.78% are achieved. Naive Bayes gives accuracy of 96.48% and k-Nearest Neighbours (kNN) gives accuracies in range of 96.34-97.06%. AdaBoost classifier achieves 96.19%, Fuzzy Unordered Role Induction algorithm achieves 96.78% and Decision Tree - J48 achieves 96.48% accuracy. If Rotation forest classifier is used in combination with Genetic Algorithm-based feature selection, accuracy of 99.48% is achieved (24-27).

All studies that are presented are showing effect of dataset size or effect of configuration on classifier performances. If an ANN is used, few questions can be asked.

- How various ANN designs perform according to dataset size?
- Is there a possibility of achieving high performances with smaller training datasets if the right ANN architecture is utilized?

In this paper, we present a study where effect of dataset size on classification performances is shown for 18 different ANN designs. According to obtained results, the ANN design with the best performances on small training datasets can be chosen. First, WBC Dataset is described. In this section, all methods used for dataset definition are described. After that, the AN is described alongside with Logistic sigmoid, Tanh and Rectified linear unit (ReLU) activation function. For evaluating classifier performances, Receiver Operating Characteristics (ROC) analysis is presented. As a classifier performance measure, area under the ROC curve (AUC) is defined. With aim of simplification, hidden layer notation method is presented. After defining all segments of this study, research methodology is described. With aim of simplification, new hidden layer notation method is presented. All obtained results are shown and commented in Results and Discussion section.

DATASET DESCRIPTION

For purpose of this research, WBC Dataset is used. Dataset is created by computing features from an image of breast mass. These images are obtained from a fine-needle aspirate. For feature extraction, a deformable spline known as a snake is used (28). The snake represents an energy minimalizing spline, where minimal energy has occurred when spline accurately corresponds to the contour. In this case, the contour represents the edge of the cell nucleus. By using the snake, active edge localization is enabled (29). Features are computed as follows (28).

1. Radius - Lengths of radial line segments defined by geometric center of a snake and individual snake points are computed. By averaging these lengths, Radius is obtained.
2. Perimeter - The Perimeter (P) is computed as a sum of all snake segments.
3. Area - Area (A) is represented as an area inside the perimeter.
4. Compactness - This feature can be computed as:

$$C = \frac{P^2}{A}, \tag{1}$$

   where C represents compactness. Irregularly shaped nucleuses will have higher and circularly shaped nucleuses will have lower Compactness.
5. Smoothness - Smoothness is computed by measuring the difference between length of a radial line and average length of neighbour lines.
6. Concavity - If a chord is drawn between two non-adjacent snake points, Concavity represents the extent to which a snake lies on the inside of the chord.
7. Concave Points - Similarly to Concavity, this feature represents a number of points that are lying inside the chord.
8. Symmetry - For symmetry measuring, the longest chord thought geometric center of snake is found. Length difference between lines, perpendicular to the longest chord, is computed in both directions.
9. Fractal Dimension - The fractal dimension of a nucleus is computed using Mandelbrot's coastline approximation.
10. Texture - The texture of a nucleus is computed as variance between pixels values.

Dataset is constructed with mean value, maximal value and standard error of each feature. In this way, a 30-dimensional dataset with 569 samples is created.

ACTIVATION FUNCTIONS

The base of every ANN is an artificial neuron (AN). The AN represents a mathematical model of a biological neuron, which is a key element of every biological nervous system (30, 31). One way to represent the AN is to use the block diagram as shown in Figure 1.

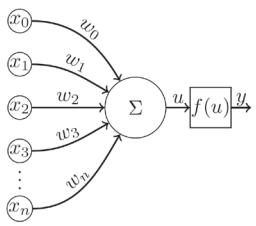

Figure 1: Block diagram of an artificial neuron

According to the diagram, AN output can be defined as:

$$y = f(u), \tag{2}$$

where

$$u = \sum_{i=0}^{n} x_i w_i. \tag{3}$$

Values of input nodes x are multiplied by weights of neuron synapses w. After multiplication, all values are summed. That value is then transformed by the activation function f(u). The activation function represents a trigger value for binary classification. This is inspired by action potential of a biologi-

cal neuron (32). In this study three different activation functions are used. These functions are: Logistic sigmoid, Tanh and ReLU. Logistic sigmoid function, along with Tanh function, represents a standard activation function used in majority of deep ANNs (32, 33).

Tanh

Tanh activation function is based on hyperbolic tangent function:

$$f(x) = \tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}, \quad (4)$$

which is defined as ratio between the hyperbolic sine and the hyperbolic cosine. Tanh activation function will produce output which falls into the interval:

$$y \in [-1,1]. \quad (5)$$

This interval represents codomain of a hyperbolic tangent function. Because of this characteristic, Tanh activation function will map value from the domain interval:

$$x \in < -\infty, +\infty >, \quad (6)$$

into the codomain interval shown in Eq. (5) (34).

Logistic sigmoid

Logistic sigmoid activation function is defined as:

$$f(x) = \frac{1}{(1+e^{-x})}. \quad (7)$$

This activation function will map all values from the domain interval:

$$x \in < -\infty, +\infty >, \quad (8)$$

into the interval:

$$y \in [0,1]. \quad (9)$$

Shown interval represents the codomain interval of Logistic sigmoid activation function.

ReLU

Rectified linear unit (ReLU) is an activation function which returns zero if x is less than zero and returns x if x is equal or greater than zero

$$f(x) = \begin{cases} 0 : x < 0 \\ x : x \geq 0 \end{cases}. \quad (10)$$

By using ReLU activation function, values from the domain interval

$$x \in < -\infty, +\infty >, \quad (11)$$

are mapped to the codomain interval

$$y \in [0, +\infty >. \quad (12)$$

Because of this property, ReLU activation function will set to zero all negative values (34).

ROC ANALYSIS

Receiver operating characteristic (ROC) analysis is a statistical method for diagnostic test evaluation. This analysis is used for determination of binary classifier accuracy. If a measure of classifier performance is defined as ratio between number of correct classifications and total number of samples, result can be misleading (36). This is typical of rare diseases and all datasets with small number of positive samples in general (37). ROC analysis is used to overcome disadvantages of this method,. Outputs of any binary classifier can be defined as 0 or 1. In case of medical diagnostic procedure, 1 marks positive and 0 marks negative medical finding. For comparison of classifier output values with true results, confusion matrix is used. For binary classifier, cofusion matrix contains two rows and two columns, illustrated in Figure 2.

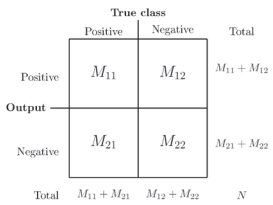

Figure 2: Confusion matrix

Element M_11 represents correct positive classification and element M_22 represents correct negative classification. On the other hand, element M_12 represents incorrect positive classification and element M_21 represents incorrect negative classification. Instead of one value which represents accuracy, with help of confusion matrix elements, two measures can be defined. Sensitivity (sn) can be defined as:

$$sn = \frac{M_{11}}{M_{11}+M_{21}}, \quad (13)$$

and it represents accuracy of classification into positive class. Additionally, specificity (sp) can be defined with:

$$sp = \frac{M_{22}}{M_{12}+M_{22}}, \quad (14)$$

and it represents accuracy of classification into negative class. With this approach, two more specific accuracy measures are given (25, 38). These two values can be used to construct the ROC curve. ROC curve is a curve on a graph where x-axis presents false positive rate (fpr) and y-axis presents a true positive rate (tpr). In this case, tpr can be defined as:

$$tpr=sn, \quad (15)$$

and fpr can be defined as:

$$fpr=1-sp. \quad (16)$$

With tpr and fpr of a classifier the point which represents classifier performance can be written as:

$$T(fpr,tpr). \quad (17)$$

By using this point together with points T_0,0 and T_1,1 an ROC curve can be constructed. Point T_0,0 represents a classifier which classifies all samples into a negative class. On the other hand, point T_1,1 represents a classifier which classifies all samples into a positive class (36). Together with point T_1,0, ROC curve makes a polygon with area under the curve (AUC). AUC is part of the interval:

$$0 \leq AUC \leq 1. \quad (18)$$

For classifier with tpr=0.95 and fpr=0.1 AUC will be 0.92 (see Figure 3).

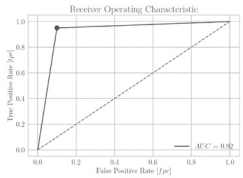

Figure 3: ROC curve for classifier with fpr=0.1 and tpr=0.95. Point T(fpr,tpr) is

marked with a dot. The dashed line represents the line of "The coin-flip classification".

The line connecting T_0,0 with T_1,1 represents random performance with AUC=0.5. This line represents "The coin-flip classification". In this case, classification will be random with expected value:

$$E[y] = 0.5,$$

according to the law of large numbers (36). All points above the line of random classification are representing classification with a certain accuracy. Points below the line of random classification are representing inverse classification (36). With other words, this classifier will perform classification into opposite classes, with certain accuracy. Classifier will perform better, if

$$\text{AUC} \rightarrow 1. \quad (19)$$

In this article AUC is used as a measure of the ANN performance.

NOTATION EXPLANATION

With the aim for simplification of ANN description, a notation method is described. This way, all hidden layers will be represented by a number of nodes in each layer inside curly brackets. For example, if a ANN with one hidden layer of 100 and one hidden layer of 20 nodes is used, a notation will be:

$$L = \{100,20\}, \quad (20)$$

where L represents configuration of hidden layers.

METHODOLOGY

For purposes of this research, 18 different ANNs are designed. ANNs are designed with each of three described activation functions and with six different hidden layer variations. These variations are:

- L_1={50},
- L_2={100},
- L_3={50,20},
- L_4={100,20},
- L_5={100,50,20} and
- L_6={100,50,50,20}.

The original dataset of 569 samples is divided into two groups. First group represent training dataset of 410 samples, and second dataset represents a test dataset of 159 samples. Each of described ANNs is trained with a dataset of size N, where

$$\overline{AUC_N} = \frac{AUC_{AdamN} + AUC_{lbfgsN} + AUC_{sgdN}}{3}. \quad (21)$$

For every dataset size N, ROC analysis of a classifier is performed and AUC value is obtained. ROC analysis is performed by using test dataset. Training is performed using three optimization algorithms. These algorithms are: Adam, LBFGS and SGD. Mean AUC of three training procedures, where each procedure uses a different optimization algorithm, can be defined as

$$N \in \mathbb{Z}: N \in [10,410]. \quad (22)$$

Using this data, a point on AUC versus dataset size graph can be drawn for every dataset size. X-axis of AUC versus dataset size diagram represents a size of dataset set used for classifier training. Y-axis represent AUC value. Point on AUC versus dataset size diagram represent AUC value which is achieved if the ANN is trained with corresponding dataset size (for example see Figure 4, where AUC of 0.95 is achieved with training dataset size of 60 samples).

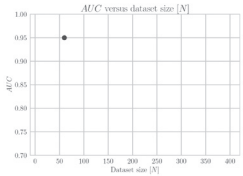

Figure 4: Example of point on AUC versus dataset size diagram for N=60 and AUC=0.95

After performing described procedure on all dataset sizes N, an AUC versus dataset size array AUC[N] is obtained, where

$$AUC[N] = [\overline{AUC_0} \quad \overline{AUC_1} \quad \ldots \quad \overline{AUC_{410}}]. \quad (23)$$

This procedure is repeated for all 18 ANNs.

RESULTS AND DISCUSSION

If AUC versus dataset size analysis is performed for the ANN with one hidden layer of 50 nodes, it can be seen that ANN designed with Logistic sigmoid, in comparison to the other two activation functions, gives higher value on all dataset sizes. By using Logistic sigmoid activation function, ANN will perform with less AUC value oscillations, trough the change of dataset size, than it is case for ReLU and Tanh activation functions (see Figure 5).

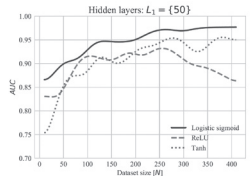

Figure 5: Comparison of AUC versus Dataset size for Logistic sigmoid, ReLU and Tanh for the ANN with one hidden layer of 50 nodes.

It can be seen that ANN designed with ReLU function, peaks on dataset size of 260 and after that falls on lower AUC values. This is not a case with the ANNs with Logistic sigmoid and Tanh activation functions. These networks have, roughly, higher AUC on larger datasets.

If performances of the ANN with one hidden layer of 100 nodes are observed, it can be seen that ANN designed with Logistic sigmoid still achieves the highest AUC on low dataset sizes, although lower than AUC of the ANN in the previous case. In this case, AUC value oscillations thought the dataset size are more pronounced if Logistic sigmoid activation function is used. For ANNs with Tanh and ReLU activation functions, these oscillations are decreased (see Figure 6).

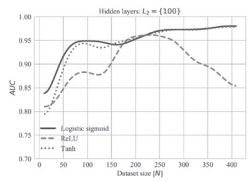

Figure 6: Comparison of AUC versus Dataset size for Logistic sigmoid, ReLU and Tanh for the ANN with one hidden layer of 100 nodes.

AUC of the ANN designed with ReLU activation function, as is it case for the ANN of {50}, peaks on dataset size of 260 and then falls. ANN designed with Tanh activation function, gives lower AUC values for smaller dataset sizes, but on higher dataset sizes gives similar AUC value as the ANN designed with Logistic sigmoid. These two ANNs are giving higher AUC value on larger datasets.

ANN with {50,20} hidden layers achieves the highest AUC values on small dataset sizes if ReLU activation function is used. Certain oscillations of AUC can be occurred for all activation function (see Figure 7).

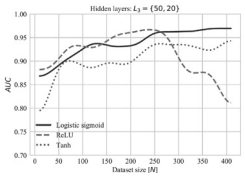

Figure 7: Comparison of AUC versus Dataset size for Logistic sigmoid, ReLU and Tanh for the ANN with one hidden layer of 50 and one hidden layer of 20 nodes.

If Logistic sigmoid is used, slightly lower AUC values are achieved with smaller datasets, but higher with larger datasets. The ANN with Tanh achieves the lowest AUC rate on smaller datasets size. When trained with larger datasets, the ANN with Tanh will, from an AUC standpoint, perform better in comparison with the ANN designed with ReLU activation function. The ANN with ReLU function achieves lower AUC value for larger datasets.

When first hidden layer of 50 is replaced by first hidden layer of 100, the ANN designed with ReLU activation function achieves AUC of 0.95 for small datasets. This characteristic can be occurred from the first dataset size of 10 samples (see Figure 8).

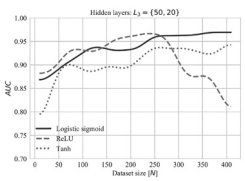

Figure 8: Comparison of AUC versus Dataset size for Logistic sigmoid, ReLU and Tanh for the ANN with one hidden layer of 100 and one hidden layer of 20 nodes.

The ANN with ReLU achieves lower AUC if it is trained with larger datasets, similar to previous cases. ANNs designed with Tanh and Logistic sigmoid are achieving lower AUC values on smaller datasets. These ANNs are also achieving higher AUC values on larger datasets.

In the case of the ANN, where hidden layer of 50 nodes is put between nodes of 100 and 20, differences between ReLU and other two networks are even more pronounced for smaller datasets. The ANN designed with ReLU activation function, achieves AUC above 0.95 for all datasets with less than 300 samples. For larger datasets, fall of AUC value can occur(see Figure 9).

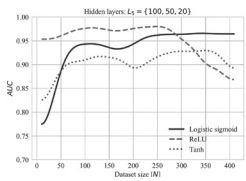

Figure 9: Comparison of AUC versus Dataset size for Logistic sigmoid, ReLU and Tanh for the ANN with one hidden layer of 100, one hidden layer of 50 and one hidden layer of 20 nodes.

The ANN designed using neurons with Logistic sigmoid, achieves greater AUC than the ANN designed with ReLU activation function, if it is trained on larger datasets. In difference of the ANN which uses Logistic sigmoid, in the case of the ANN designed with Tanh activation function, fall of AUC value on larger datasets can be noticed.

When another hidden layer of 50 nodes is added, forming than {100,50,50,20} hidden layer configuration, the ANN with ReLU activation function is achieving AUC greater than 0.95 for all datasets with less than 300 samples. For larger datasets, the ANN with ReLU is achieving AUC of 0.94 almost constantly (see Figure 10). According to these results, ANN with {100,50,50,20} hidden layer configuration and ReLU activation function can achieve high AUC value, irregardless of dataset size.

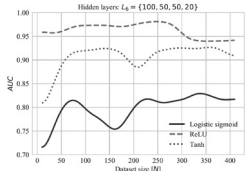

Figure 10: Comparison of AUC versus Dataset size for Logistic sigmoid, ReLU and Tanh for the ANN with one hidden layer of 100, two hidden layers of 50 and one hidden layer of 20 nodes.

ANNs designed with Tanh and Logistic sigmoid are, in case of this configuration of hidden layers, achieving considerably lower AUC values for all dataset sizes.

Presented results show that deeper networks with ReLU activation function achieves higher AUC values on smaller datasets. For all larger datasets, highest AUC values are achieved if ANNs with smaller number of hidden layers and Logistic sigmoid activation function are used. Best results are achieved if ANN with configuration {100,50,50,20} and ReLU activation function is used for datasets in range from 10 to 282 samples. For dataset sizes larger than 282 samples, best results are achieved if ANN with configuration {50} and Logistic sigmoid activation function is used (see Figure 11).

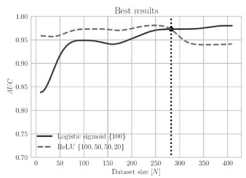

Figure 11: AUC versus dataset size of the ANN with the best performance before demarcation line and AUC versus dataset size of the ANN with the best performance after demarcation line.

Demarcation line between two ANNs is marked with dotted line. This line presents minimal dataset size on which better performances are achieved using ANN with Logistic sigmoid activation function and less hidden layers. For smaller datasets, better results are achieved using deeper ANNs with ReLU activation function. In case of WBC Dataset, demarcation line is set to dataset size of 282 samples. According to shown results, the optimal activation func-

tion can be written as:

$$f(x,N)=\begin{cases}0: & x<0\\ x: & x\geq 0\end{cases}: \quad N<282$$
$$\frac{1}{1+e^{-x}}: \quad N\geq 282. \tag{24}$$

In addition to activation function, configuration of hidden layers can be written as:

$$f(x,N)=\begin{cases}0: & x<0\\ x: & x\geq 0\end{cases}: \quad N<282$$
$$\frac{1}{1+e^{-x}}: \quad N\geq 282. \tag{25}$$

CONCLUSION

In this study, we have shown dependence of ANN performance on dataset size. According to our results, it can be noticed that various ANN designs have different AUC versus dataset size performance. For smaller datasets, in range between 10 and 282 samples, the best performances are achieved using deep ANNs with ReLU activation function. These designs offer AUC values larger than 0.95 for all dataset sizes. Furthermore, deep ANNs with ReLU offer stable AUC value with low oscillations for all datasets in range between 10 and 282 samples. On the other hand, if larger datasets are used for training, best results are achieved if ANNs with smaller number of hidden layers and Logistic sigmoid activation function are used. These ANNs offer better performances with larger training datasets. Our study shows how ANNs perform when trained with small datasets. In practice, large datasets can often be unavailable for classifier training, therefore it is important to design the classifier that has maximal AUC, which is possible to achieve with the available data.

ACKNOWLEDGEMENTS

This research has been (partly) supported by the CEEPUS network CIII-HR-0108, European Regional Development Fund under the grant KK.01.1.1.01.0009 (DATACROSS), project CEKOM under the grant KK.01.2.2.03.0004 and University of Rijeka scientific grant uniri-tehnic-18-275-1447.

DISCLOSURE STATEMENT

The authors declare no conflict of interest, financial or otherwise.

REFERENCES

1. Lorencin I, Anđelić N, Mrzljak V, Car Z. Genetic Algorithm Approach to Design of Multi-Layer Perceptron for Combined Cycle Power Plant Electrical Power Output Estimation. Energies. 2019 Jan;12(22):4352.
2. Baressi Šegota S, Anđelić N, Lorencin I, Saga M, Car Z. Path planning optimization of six-degree-of-freedom robotic manipulators using evolutionary algorithms. International Journal of Advanced Robotic Systems. 2020 Mar 19;17(2):1729881420908076.
3. Baressi Šegota S, Lorencin I, Ohkura K, Car Z. On the Traveling Salesman Problem in Nautical Environments: an Evolutionary Computing Approach to Optimization of Tourist Route Paths in Medulin, Croatia. Journal of Maritime & Transportation Sciences. 2019 Jul 1;57(1).
4. Brnić M, Čondrić E, Blažević S, Anđelić N, Borović E, Car Z. Sepsis prediction using artificial intelligence algorithms. InInternational Conference on Innovative Technologies, IN-TECH 2018 2018 Jan 1.
5. Ramesh AN, Kambhampati C, Monson JR, Drew PJ. Artificial intelligence in medicine. Annals of the Royal College of Surgeons of England. 2004 Sep;86(5):334.
6. Baressi Šegota S, Anđelić N, Kudláček J, Čep R. Artificial neural network for predicting values of residuary resistance per unit weight of displacement. Journal of Maritime & Transportation Sciences. 2019 Jul 1;57(1).
7. Pereira F, Mitchell T, Botvinick M. Machine learning classifiers and fMRI: a tutorial overview. Neuroimage. 2009 Mar 1;45(1):S199-209.
8. Bogović K, Lorencin I, Anđelić N, Blažević S, Smolčić K, Španjol J, Car Z. Artificial intelligence-based method for urinary bladder cancer diagnostic. InInternational Conference on Innovative Technologies, IN-TECH 2018 2018 Jan 1.
9. Lorencin I, Anđelić N, Španjol J, Car Z. Using multi-layer perceptron with Laplacian edge detector for bladder cancer diagnosis. Artificial Intelligence in Medicine. 2020 Jan 1;102:101746.
10. Kulik SD. NEURAL NETWORK MODEL OF ARTIFICIAL INTELLIGENCE FOR HANDWRITING RECOGNITION. Journal of Theoretical & Applied Information Technology. 2015 Mar 20;73(2).
11. Bui V, Chang LC. Deep learning architectures for hard character classification. InProceedings on the International Conference on Artificial Intelligence (ICAI) 2016 (p. 108). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).
12. Sladojevic S, Arsenovic M, Anderla A, Culibrk D, Stefanovic D. Deep neural networks based recognition of plant diseases by leaf image classification. Computational intelligence and neuroscience. 2016;2016.
13. Hirasawa T, Aoyama K, Tanimoto T, Ishihara S, Shichijo S, Ozawa T, Ohnishi T, Fujishiro M, Matsuo K, Fujisaki J, Tada T. Application of artificial intelligence using a convolutional neural network for detecting gastric cancer in endoscopic images. Gastric Cancer. 2018 Jul 1;21(4):653-60.
14. Horie Y, Yoshio T, Aoyama K, Yoshimizu S, Horiuchi Y, Ishiyama A, Hirasawa T, Tsuchida T, Ozawa T, Ishihara S, Kumagai Y. Diagnostic outcomes of esophageal cancer by artificial intelligence using convolutional neural networks. Gastrointestinal endoscopy. 2019 Jan 1;89(1):25-32.
15. Kim DH, MacKinnon T. Artificial intelligence in fracture detection: transfer learning from deep convolutional neural networks. Clinical radiology. 2018 May 1;73(5):439-45.
16. Lu H, Li Y, Uemura T, Ge Z, Xu X, He L, Serikawa S, Kim H. FDCNet: filtering deep convolutional network for marine organism clas-

sification. Multimedia tools and applications. 2018 Sep 1;77(17):21847-60.
17. Nachtigall LG, Araujo RM, Nachtigall GR. Classification of apple tree disorders using convolutional neural networks. In2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI) 2016 Nov 6 (pp. 472-476). IEEE.
18. Nguyen KA, Sahin O, Stewart RA, Zhang H. Smart technologies in reducing carbon emission: Artificial intelligence and smart water meter. InProceedings of the 9th International Conference on Machine Learning and Computing 2017 Feb 24 (pp. 517-522).
19. Wang S, Zhang Y, Ji G, Yang J, Wu J, Wei L. Fruit classification by wavelet-entropy and feedforward neural network trained by fitness-scaled chaotic ABC and biogeography-based optimization. Entropy. 2015 Aug;17(8):5711-28.
20. Capizzi G, Sciuto GL, Napoli C, Tramontana E, Woźniak M. Automatic classification of fruit defects based on co-occurrence matrix and neural networks. In2015 Federated Conference on Computer Science and Information Systems (FedCSIS) 2015 Sep 13 (pp. 861-867). IEEE.
21. Prusa J, Khoshgoftaar TM, Seliya N. The effect of dataset size on training tweet sentiment classifiers. In2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA) 2015 Dec 9 (pp. 96-102). IEEE.
22. Ohno-Machado L, Fraser HS, Ohrn A. Improving machine learning performance by removing redundant cases in medical data sets. InProceedings of the AMIA Symposium 1998 (p. 523). American Medical Informatics Association.
23. Foody GM, McCulloch MB, Yates WB. The effect of training set size and composition on artificial neural network classification. International Journal of Remote Sensing. 1995 Jun 1;16(9):1707-23.
24. Tsangaratos P, Ilia I. Comparison of a logistic regression and Naïve Bayes classifier in landslide susceptibility assessments: The influence of models complexity and training dataset size. Catena. 2016 Oct 1;145:164-79.
25. Tariq N. Breast cancer detection using artificial neural networks. J Mol Biomark Diagn. 2017;9(371):2.
26. Gbenga DE, Christopher N, Yetunde DC. Performance comparison of machine learning techniques for breast cancer detection. Nova. 2017;6(1):1-8.
27. Aličković E, Subasi A. Breast cancer diagnosis using GA feature selection and Rotation Forest. Neural Computing and Applications. 2017 Apr 1;28(4):753-63.
28. Obaid OI, Mohammed MA, Ghani MK, Mostafa A, Taha F. Evaluating the performance of machine learning techniques in the classification of Wisconsin Breast Cancer. International Journal of Engineering & Technology. 2018;7(4.36):160-6.
29. Joshi A, Ashish M. Analysis Of K-Nearest Neighbor Technique For Breast Cancer Disease Classification. Int. J. Recent Sci. Res. 2017;8(8):1005-9008.
30. Street WN, Wolberg WH, Mangasarian OL. Nuclear feature extraction for breast tumor diagnosis. InBiomedical image processing and biomedical visualization 1993 Jul 29 (Vol. 1905, pp. 861-870). International Society for Optics and Photonics.
31. Sibi P, Jones SA, Siddarth P. Analysis of different activation functions using back propagation neural networks. Journal of Theoretical and Applied Information Technology. 2013 Jan 31;47(3):1264-8.
32. Kass M, Witkin A, Terzopoulos D. Snakes: Active Contour Models‖ International Journals of Computer Vision.
33. Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. InProceedings of the fourteenth international conference on artificial intelligence and statistics 2011 Jun 14 (pp. 315-323).
34. Lorencin I, Anđelić N, Mrzljak V, Car Z. Marine Objects Recognition Using Convolutional Neural Networks. NAŠE MORE: znanstveno-stručni časopis za more i pomorstvo. 2019 Sep 26;66(3):112-9.
35. Karlik B, Olgac AV. Performance analysis of various activation functions in generalized MLP architectures of neural networks. International Journal of Artificial Intelligence and Expert Systems. 2011 Feb;1(4):111-22.
36. Fawcett T. An introduction to ROC analysis. Pattern recognition letters. 2006 Jun 1;27(8):861-74.
37. Metz CE. Basic principles of ROC analysis. InSeminars in nuclear medicine 1978 Oct 1 (Vol. 8, No. 4, pp. 283-298). WB Saunders.
38. Lorencin I, Anđelić N, Mrzljak V, Car Z. Multilayer Perceptron approach to Condition-Based Maintenance of Marine CODLAG Propulsion System Components. Pomorstvo. 2019 Dec 19;33(2):181-90.