

## INFLUENCE OF HIDDEN NEURON NUMBER ON THE PERFORMANCE OF ANN MODELS APPLIED TO DEINKING FLOTATION DATA

Katarina Balanović<sup>1a</sup>, Maja Trumić<sup>1b</sup>, Tamara Gavrilović<sup>2a</sup>

<sup>1</sup> Technical Faculty in Bor, University of Belgrade, V.J. 12, 19210 Bor, Serbia

<sup>2</sup> Faculty of Technical Sciences, University of Pristina, Knjaza Miloša 7, 38220 Kosovska Mitrovica, Serbia

<sup>1a</sup> [kbalanovic@tfbor.bg.ac.rs](mailto:kbalanovic@tfbor.bg.ac.rs), 0000-0002-2197-5718

<sup>1b</sup> [majatrumic@tfbor.bg.ac.rs](mailto:majatrumic@tfbor.bg.ac.rs), 0000-0001-9361-4412

<sup>2a</sup> [tamara.gavrilovic@pr.ac.rs](mailto:tamara.gavrilovic@pr.ac.rs), 0000-0003-3448-7753

### Abstract

*In this paper, the influence of the optimal number of neurons in an artificial neural network model, applied to data obtained from deinking flotation is presented and analyzed. The results indicate that when a well-performing model is selected, it is not necessary to search for the optimal number of hidden neurons. However, when the applied model does not yield satisfactory results, determining the optimal number of neurons becomes essential.*

**Keywords:** Artificial neural networks, Hidden neurons, Flotation, Deinking.

### 1. INTRODUCTION

Artificial intelligence is one of the fastest-developing fields in computer science in recent decades [1]. It represents a discipline within computer science dedicated to creating programs that allow machines to act and make decisions in ways that resemble intelligent behavior. In this context, intelligent thinking refers to the capacity to reason and respond appropriately to new, previously unknown situations [2]. Machine learning is a subfield of artificial intelligence concerned with developing programs that can learn from previously collected data. Once a solution to a problem is identified, the program is able to store this knowledge and apply it to similar situations. Machine learning is generally categorized into three main types: supervised learning, unsupervised learning, and reinforcement learning [3]. In supervised learning, the program is provided with a set of input data ( $x_1, x_2, \dots, x_n$ ) along with the corresponding desired output values, where each input  $x_i$  is associated with the correct output  $y_i$ . The goal of the program is to learn how to correctly predict the output value for new, unseen input data. Algorithms that use supervised learning include artificial neural networks, decision trees, support vector machines, Gaussian processes, among others [3].

An Artificial Neural Network (ANN) is a system of interconnected simple processing elements (neurons) inspired by the functioning of biological neurons. A typical ANN architecture is organized into three types of layers: input, hidden, and output layers [2,3]. The optimal number of hidden neurons depends on multiple factors, including the number of inputs and outputs, number of training pairs, level of noise in the training pairs, complexity of the error function, network architecture, and training algorithm. It can be said that the choice of architecture depends on the type and complexity of the problem being studied [3].

A precise and reliable method for determining the minimum required number of neurons has not yet been established. What can be estimated, to some extent, is the upper limit, that is, the maximum number of neurons in the hidden layer that can be used for modeling a system represented by a given dataset. The most common approach to determining the number of neurons in hidden layers is to train multiple networks with varying neuron counts, gradually increasing the

number of neurons until the desired result is achieved [3,4]. The determination of the optimal number of hidden layer neurons is still most commonly carried out experimentally, using the trial-and-error approach [5]. In this case, it is recommended to start with the minimal structure, i.e., a network with a single neuron in the hidden layer, and then gradually increase the number of neurons while evaluating the obtained results. The main drawback of this method lies primarily in the large number of networks that need to be trained and tested [3].

The optimal number of neurons in the hidden layer is very important for the correct functioning of the model. A network with too many neurons will have an excessive number of parameters, making it prone to overfitting. Conversely, a network with too few neurons will not be able to properly approximate the given nonlinear relationships, leading to a problem of underfitting [3].

The flotation process has been used in mineral processing plants to separate valuable minerals from the gangue. This process takes place in a specific environment – flotation pulp – which consists of three phases: solid (mineral particles), liquid (water) and gaseous (air). Separation of mineral particles is made possible due to the differences in their surface hydrophobicity (feature of material characterizing its ability to be wetted with a liquid in the presence of a gas phase) [6]. Compared to mineral flotation, deinking flotation (ink removal from cellulose fibers) is a relatively young process. The basis of deinking flotation is pulp that contains the following components: water, as the carrier medium (approximately 98-99%); fibers, material obtained as a clean fraction (1-2%); fillers (<0.6%), printing ink and other impurities (<0.15%) that need to be removed; and reagents required for the fiber purification process (<0.1%) [7].

The efficiency of the flotation process can be shown through the toner recovery in foam product and cellulose fiber recovery in sink product, effective residual ink concentration in the final product (ERIC), the whiteness and brightness of the fibers. The main parameters that influence flotation are temperature, pH, toner particle size, air flow, water hardness (concentration of calcium ions in the water), and others. All parameters are interdependent, so the flotation process is defined exclusively by their interaction-i.e., by the mechanism of their combined effect [8,9]. Our understanding of the fundamental physical and chemical processes underlying the deinking processes is growing. But, due to the great variability of the raw material as well as the complexity of the physical processes which occur at the microscopic level, processing problems and process instability remain as challenges [10].

In the following study, an algorithm with supervised learning was analyzed, specifically, a model based on artificial neural networks for data obtained by deinking flotation.

## **2. EXPERIMENTAL**

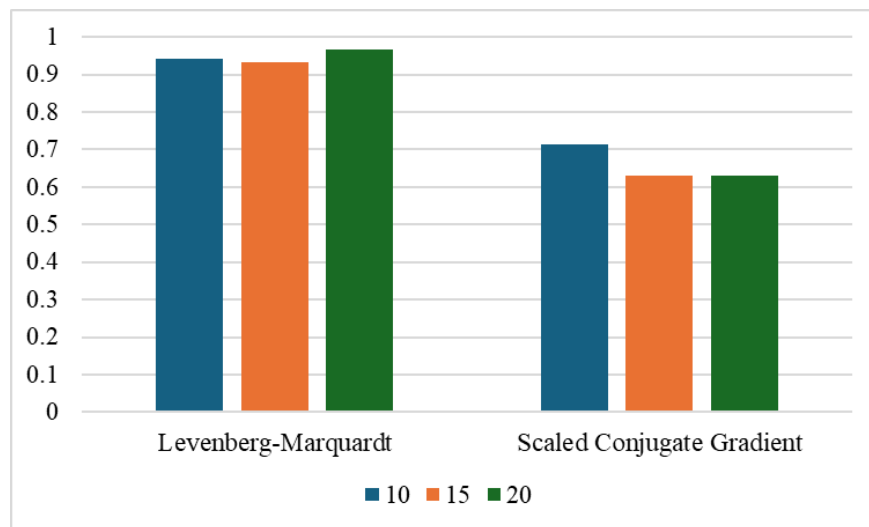
The Neural Fitting tool from the Matlab software package was used to create and train the network, as well as to evaluate its performance accuracy using mean squared error (MSE) and regression analysis. The time of deinking flotation (in the range 1-20 min), pH value (in the range 3-12), reagents such as oleic acid, oleic acid with  $\text{CaCl}_2$ , oleic acid with  $\text{AlCl}_3$  as well as concentration of reagents (0.1 kg/t, 0.4 kg/t, 1 kg/t) were used as input model variables. The recovery of toner particles in froth was used as the output model parameters. From the total dataset of 225 parameters, 200 (88.89%) were randomly allocated for network training, validation, and testing, while the remaining 25 (11.11%) were reserved for evaluating the trained network's accuracy.

## **3. RESULTS AND DISCUSSION**

The neural network was created, trained, and tested with 10, 15, and 20 hidden neurons. From the available dataset of 200 parameters, 140 (70%) were used for training, while 30 parameters each (15%) were allocated for validation and testing (Table 1, Figure 1). The training was performed using the Levenberg–Marquardt and Scaled Conjugate Gradient algorithms.

*Table 1. Network performance accuracy obtained with the Levenberg–Marquardt and Scaled Conjugate Gradient algorithms*

| Training algorithm        | Number of neurons in hidden layer | Mean squared error (MSE) | Coefficient of correlation (R) |
|---------------------------|-----------------------------------|--------------------------|--------------------------------|
| Levenberg-Marquardt       | 10                                | 136.83548                | 0.942087                       |
|                           | 15                                | 275.72803                | 0.932717                       |
|                           | 20                                | 269.38345                | 0.967081                       |
| Scaled Conjugate Gradient | 10                                | 208.42001                | 0.713716                       |
|                           | 15                                | 253.86608                | 0.630678                       |
|                           | 20                                | 251.27352                | 0.629073                       |



*Figure 1. Comparison of correlation coefficients for neural networks trained with the Levenberg–Marquardt and Scaled Conjugate Gradient algorithms*

Testing of the neural network trained with the Levenberg–Marquardt algorithm achieves good results, as shown by a correlation coefficient above 0.9 [11]. In contrast, the neural network trained with the Scaled Conjugate Gradient algorithm did not provide satisfactory results, as reflected in a lower correlation coefficient. Moreover, for the network trained with the Scaled Conjugate Gradient algorithm, the correlation coefficient decreased with an increasing number of hidden neurons, which may indicate overfitting. This issue was not observed in the network trained with the Levenberg–Marquardt algorithm.

From the above, it can be concluded that achieving good results depends more on selecting the optimal training algorithm, while in cases where the optimal model cannot be used, it is necessary to determine the optimal number of neurons in the hidden layer for the given dataset. The required number of neurons should be determined iteratively, starting with a network containing a single neuron in the hidden layer, and then gradually increasing the number of neurons until satisfactory results are obtained.

The greater the number of neurons, the more complex the function that can be approximated. On the other hand, a higher number of neurons requires more data, which may also lead to overfitting. The obtained neural networks can be reset or initialized with new initial weights and retrained; the number of neurons can be reduced; the training dataset can be expanded; the number of input values can be adjusted; the ratio of data used for training, validation, and testing can be modified; alternative learning and training algorithms can be applied; and preprocessing of inputs and outputs can be performed [10].

#### 4. CONCLUSION

In this study, supervised learning algorithms were analyzed, with a focus on artificial neural network models applied to data obtained from deinking flotation. The neural network architecture consisted of a single hidden layer with 10, 15, and 20 neurons. The networks were created, trained, and tested using the Levenberg–Marquardt and Scaled Conjugate Gradient algorithms.

The inputs of the models were time of deinking flotation, pH, reagents such as oleic acid, oleic acid with  $\text{CaCl}_2$ , oleic acid with  $\text{AlCl}_3$  as well as concentration of reagents, while the output values were recovery of toner particles in froth.

Testing of the neural network trained with the Levenberg–Marquardt algorithm achieved better results compared to the Scaled Conjugate Gradient algorithm. Based on the obtained results, it can be concluded that achieving good performance depends more on the choice of an optimal training algorithm, while in cases where the optimal model cannot be applied, it is necessary to determine the optimal number of hidden neurons for the given dataset. Since this study used 10, 15, and 20 hidden neurons in the neural network architecture, determining and applying the optimal number of neurons for this specific dataset could potentially yield even better results.

#### ACKNOWLEDGEMENTS

*The research presented in this paper was done with the financial support of the Ministry of Science, Technological Development and Innovation of the Republic of Serbia, within the funding of the scientific research work at the University of Belgrade, Technical Faculty in Bor, according to the contract with registration number 451-03-137/2025-03/ 200131.*

#### REFERENCES

- [1] J.Đ. Novaković., Rešavanje klasifikacionih problema mašinskog učenja, Fakultet tehničkih nauka u Čačku, 2013.
- [2] D. Tanikić., Veštačke neuronske mreže, fazi logika i genetski algoritam, Univerzitet u Beogradu, Tehnički fakultet u Boru, 2016.
- [3] M.M. Kovačević., Model za prognozu i procenu troškova izgradnje armirano-betonskih drumskih mostova, Doctoral dissertation, University of Belgrade, Serbia, 2018.
- [4] M. Kovačević, N. Ivanišević, T. Dašić, Lj. Marković., *Gradjevinar*, 1 (2018) 1-10.
- [5] H.R. Maier, G.C. Dandy., *Envir. Model. Softw.*, 15 (2000) 101-124.
- [6] N. Čalić., *Teorijski osnovi pripreme mineralnih sirovina*, Faculty of Mining and Geology, University of Belgrade, 1990, 496, (in Serbian).
- [7] M. Kemper., *Int. J. Miner. Process.*, 56 (1-4) (1999) 317-333.
- [8] J.K. Borchardt., *Colloids Surf. A Physicochem. Eng. Asp.*, 88 (1) (1994) 13-25.
- [9] R. Svensson., *The influence of surfactant chemistry on flotation deinking*, 2011.
- [10] W. J. Pauck., *Neural network modelling and prediction of the flotation deinking behaviour of complex recycled paper mixes*, Doctoral dissertation, Faculty of Engineering, University of KwaZulu Natal, Durban, 2011.
- [11] M. Trumić, K. Balanović, T. Gavrilović., *Prediction of results of floatation process using artificial neural networks*, in *Proceedings of XVI International Mineral Processing and Recycling Conference*, Belgrade, Serbia, 2025, 659-664.