

Article

Conveyor Belt Damage Detection with the Use of a Two-Layer Neural Network

Agata Kirjanów-Błażej * and Aleksandra Rzeszowska

Faculty of Electronics, Wrocław University of Science and Technology, Janiszewskiego 11/17, 50-372 Wrocław, Poland; aleksandra.rzeszowska@pwr.edu.pl

* Correspondence: agata.kirjanow-blazej@pwr.edu.pl

Abstract: Non-invasive conveyor belt diagnostics in damage detection allows significant reductions of the costs related to belt replacement, as well as the evaluation of belt usability and wear degree changes over time. As a result, it increases safety in the location where the belt is used. Depending on the location of a belt conveyor, its length or the type of the transported material, the belt may undergo wear at different rates, albeit the wear process itself is inevitable. This article presents an artificial intelligence-based approach to the classification of conveyor belt damage. A two-layer neural network was implemented in the MATLAB programming language, with the use of a Deep Learning Toolbox set. As a result of the optimization of the created network, the effectiveness of operation was at the level of 80%.

Keywords: neural networks; conveyor belts; magnetic method; diagnostics; automatic detection



Citation: Kirjanów-Błażej, A.; Rzeszowska, A. Conveyor Belt Damage Detection with the Use of a Two-Layer Neural Network. *Appl. Sci.* **2021**, *11*, 5480. <https://doi.org/10.3390/app11125480>

Academic Editor: Fabio La Foresta

Received: 18 May 2021

Accepted: 11 June 2021

Published: 13 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The life of the conveyor belt depends on many factors collected in the literature [1]—among other things, it depends on the type of material transported, the specificity of the transport point as well as the length and age of the conveyor belt. Figure 1 shows the appearance of a conveyor belt used in mining. This device works continuously, transporting the material from the tail pulley to the head pulley. During the operation of the belt conveyor, the belt is passed through both pulleys, changing the running direction on them. Between the pulleys, along its entire length, the belt is supported by sets of idlers. Above the conveyor belt, there is a feed chute, the arrangement and shape of which guarantees the correct arrangement of the discharged material on the conveyor belt and prevents it from spilling out of the belt area. The material is moved along the belt to the head pulley, where the belt direction changes, and the load is thrown outside the area of the belt conveyor [2].

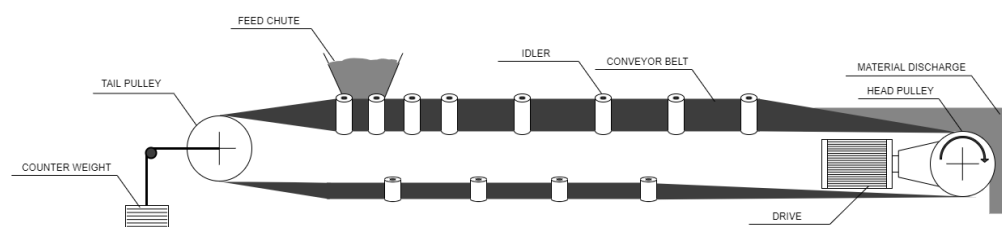


Figure 1. Diagram of the conveyor.

The most important and most frequently damaged part of the conveyor is the conveyor belt. It is estimated that its cost is about 60% of the cost of the entire conveyor [3,4]. Demands for conveyor belts require that the belt be a high-quality product, which in turn translates into its cost. The more important is its diagnosis and quick detection of potential damage when it is possible to remove them because the potential failure of the belt conveyor generates a cost associated not only with its repair, but also associated with

forced downtime in transport [3]. Figure 2 shows a diagram of a cross section through a conveyor belt.

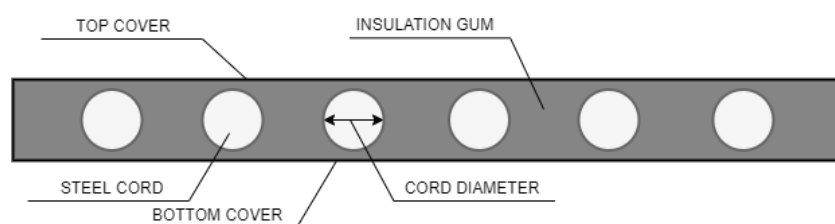


Figure 2. Diagram of the cross section through the conveyor belt.

The specificity of NDT (non-destructive testing) assumes that, during the inspection of the tested object (here the conveyor belt), it is not subject to degradation or its structure or properties are not changed. Many researchers around the world have developed many systems for the diagnosis of the core of the conveyor belts [3]. Some of the available methods are designed to diagnose the condition of covers, others can detect damage to the steel core embedded in the rubber[5,6]. In the era of better possibilities of Industry 4.0 data, it is necessary to install the sensor on the tested object and then collect data and process it.

One of the methods for non-invasive belt diagnostics is the application of a device which uses the changes in the magnetic field as the belt passes under a measurement bar (installed across the entire width of the belt). Such a device is researched at the Belt Conveying Laboratory, Wrocław University of Science and Technology (Figure 3) [7,8].



Figure 3. The device for non-invasive magnetic diagnostic of the belt.

The operation of magnetic methods in identifying damage is based on the change in the magnetic field generated by the previously magnetized cords that make up the steel core of the tape. Changes in the magnetic field may be caused by joining successive sections of the tape or by damage to the lines (cut, corrosion, missing lines). As part of this method, a magnetic field with a sufficiently large induction flux is induced in the tested object and magnetic scattering fields are searched for [9].

Based on the data obtained from the Diagbelt system, analyses were carried out [10–12] to aid the detection of defects or to select the best set of parameters (belt speed, distance from the head, sensitivity) that should be set before the system starts measurements.

The data obtained from the Diagbelt can be visualized in a two-dimensional drawing. An example of such a plot is shown in Figure 4.

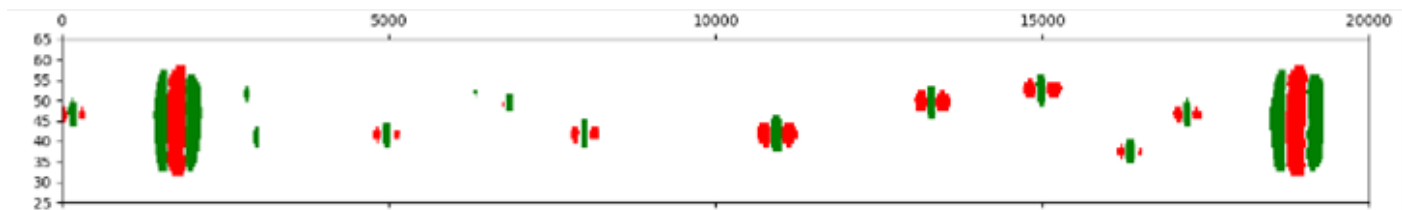


Figure 4. Representative conveyor belt—magnetic analysis of damage.

1.1. Neural Network

One inspiration for developing artificial neural networks came from biological phenomena observed in human brain. Research into the secrets of human intelligence provided results that proved useful in computer science as well. Neural networks, so often used in solving technical problems, are then in fact a simplified model of the human nervous system [13–15]. An artificial neural network is a structure composed of single calculation units and consisting of one or many layers processing a certain output signal on the basis of a particular input signal. The input layer of a neural network consists of a number of neurons corresponding to the number of values provided at the network input (occasionally, in order to improve its performance, the network is provided with an additional neuron activated by a constant signal). The hidden layer may consist of any number of neurons, each of them being activated by the signal produced by the neurons in the previous layer. The output layer must consist of a number of neurons corresponding to the number of outputs in the designed network [13,16,17].

Figure 5a shows the structure of a single neuron, which receives many input values and which produces a single output value. Figure 5b shows a representative network that consists of a hidden layer with k_1 neurons. A single neuron works by adding the stimulation signal delivered to its input after multiplying it by the weights assigned to a given input. The summed excitation signal is then transferred to the appropriate activation function, and the value generated by it is the output signal of the neuron [17]. The output signal of the neuron is described by a mathematical relationship represented by the Formula (1):

$$y_i = f\left(\sum_{j=1}^N w_{ij}x_j + w_{i0}\right) \quad (1)$$

where y_i is the output signal of the i -th neuron, N is the number of signals that stimulate the neuron, w_{ij} is the weight of the i -th neuron ascribed to the j -th stimulation, and f is the activation function.

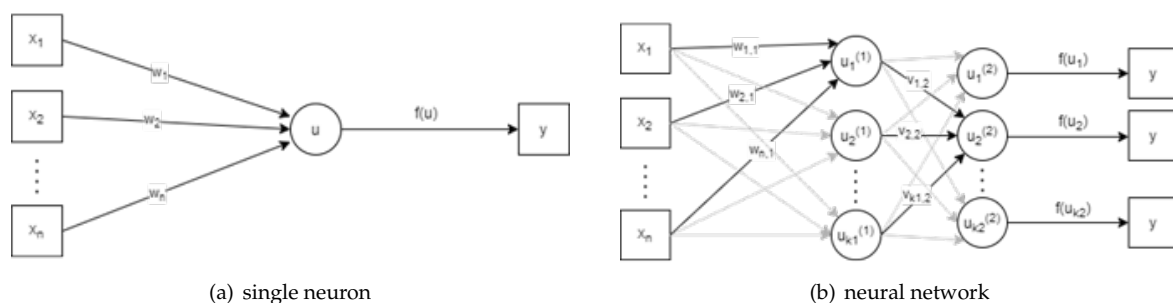


Figure 5. Models.

The number of layers and neurons in the hidden layers is one of the most important parameters of the designed neural network. Networks with one or two hidden layers are among the most frequently used (problems which would require more than three layers to solve are not known) [17,18].

The learning process of a neural network requires defining a method for updating the weights of the connections between neurons in the adjacent layers. One of the most

frequently used learning algorithms is backward error propagation, which consists of finding the minimum sum squared error (in the steepest descent optimization method). In this algorithm, the network learning error is sent from the output layer to the input layer [14].

1.2. Selection of the Size of the Neural Network

In order to best illustrate the ability of the neural network to recognize certain types of tasks set before it, it is worth considering the problem of divining the belonging of a given pair of points to a certain area by the neural network on the basis of the values given at its input. Due to the fact that it is easy to visualize the described issues in an accessible way for the human eye (on the chart), we will consider the problem with two variables, thanks to which it will be possible to plot 2D plots. While considering the operation of neural networks, there are three types of problems [17,19]:

- linearly separable problem,
- an almost linearly separable problem,
- nonlinearly separable problem.

Each of the problems named above has an appropriate interpretation used in the following considerations. In each of these problems, a certain area is designated where the points belonging to a given class are located (in the analyzed case, this area is limited by straight lines; in the case of larger problems, this area is limited by hyperplanes). Thus, in the case of a linearly separable problem, points above a certain line belong to a given class, below the same line they do not belong; in the almost linearly separable problem, the segment of a circle with the center at point $O = (0, 1)$, and radius $r = 0.5$ was determined, while, in the nonlinearly separable problem, the area is determined by two segments of a circle with centers at points $O_1 = (0, 1)$, $O_2 = (1, 0)$ and radius $r_1 = 0.7$, $r_2 = 0.5$.

There are randomly selected 1000 points in the training set, the excitation values of which were given as input from the uniform distribution $U(0, 1)$. Then, the process of teaching the neural network was carried out, followed by testing the network, in which pairs of points (x, y) with values from the uniform distribution $U(0, 1)$ were generated, and then the network response to the input of these two was checked. Figure 6 shows the points distinguished by the network response—green points mean values considered by the network to belong to a given area, red points mean values not belonging to this area. The black line shows the expected course of the network decision boundary.

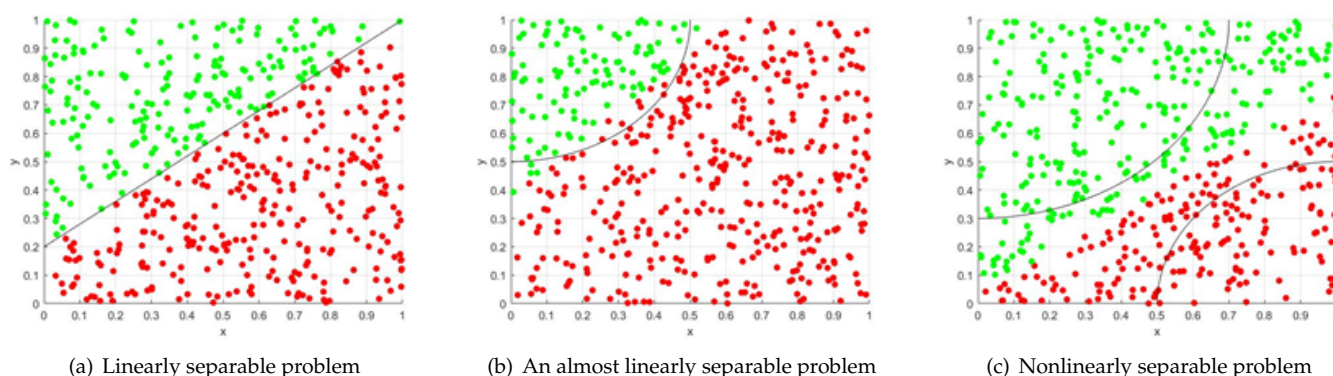


Figure 6. Single-layer network.

A single-layer network can separate the decision area only with a straight line (or a hyperplane in more dimensions). Problems of a similar type given on a two-layer network with a hidden layer with five neurons generate the same results presented in Figure 7.

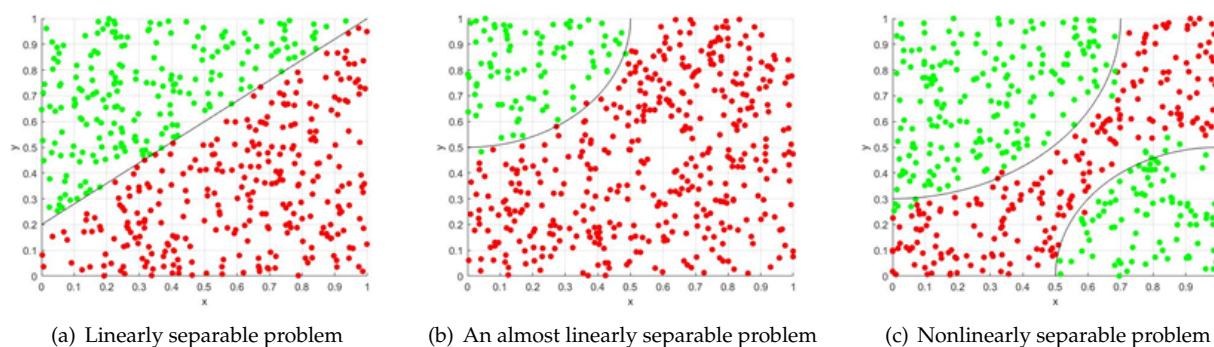


Figure 7. Two-layer network.

It can be noticed that the single-layer neural network is able to solve only the simplest problems, and, therefore, it has little practical significance. This is used only when the existence of one layer is sufficient to solve a specific problem, and the selection of the architecture of such a network is simple—the number of neurons in the input layer is clearly defined by the dimension of the input vector, and the number of output neurons is determined by the dimension of the expected output vector [16,17].

Choosing the architecture of a two-layer network is a bit more complicated. The number of neurons in the input and output layers is naturally determined, as in the case of a single-layer network, by the size of the input and output vector, but the number of neurons in the hidden layer leaves a lot of room for maneuvering to ensure matching with the training data and maintaining good generalization capabilities. Such a theoretical solution to the problem of matching to learning data has been the subject of many studies by mathematicians dealing with the approximation of functions of many variables. It can be noticed that the neural network is a kind of universal approximator of the learning data—during training, appropriate coefficients are selected (represented by the weight vectors of individual neurons), and in the reproductive process of determining the value based on the given input and the learned weight matrix, the value of the approximating function is calculated [16].

In the case of a desire to use a multilayer network (with one or more hidden layers), the problem of appropriate selection of the number of hidden layers arises. This problem is also based on the properties of approximating functions, and the solution to this problem is provided by the Kolmogorov theory, on the basis of which it can be stated that, with N input neurons, it is enough to use $(2N + 1)$ neurons in the hidden layer to solve a given approximation problem [16,20].

2. Material and Methods

Before they are fed at the input of the neural network, the data must be subjected to preliminary processing. Images fed in the form of values given to successive pixels are mainly performed with deep neural networks, which require a large base of input data in their learning process. An image 480×600 px in size fed in a classic pixel-by-pixel version requires a network with a structure of 240,000 neurons in the input layer. However, this problem may be solved, and the learning results may be satisfactory even with a small training database. In the case of this research into damage detection in a conveyor belt, the Python programming language was used to define areas located in close vicinity to each other. Figure 8 shows the borders of such areas.



Figure 8. Representative conveyor belt—borders of potentially damaged areas.

In order to avoid providing the network with the entire set of data vectors, the detected areas were used to identify three values describing the surface areas of sub-areas forming the detected area (the surface of the red area, the surface of the green area, the surface of the red area or the surface of the green area, the surface of the red area, the surface of the green area) and three values identifying the number of channels on which the signal related to a given sub-area has been read. In a situation when the detected area consists of one or two sub-areas, the missing sub-area lub liczba kanałów has the surface area or the number of channels on which the failure was detected equal to 0. In effect, a large input vector was replaced with six values. The detectable types of damage to the conveyor belt include: one cord missing, two cords missing, three cords missing, strand/wire cut, one cord partially cut, one cord cut, two cords cut, three cords cut, and belt splice.

For the so-prepared input data and possible network outputs, the structure of the neural network must comprise 6 neurons in the input layer and 9 neurons in the output layer.

3. Result and Discussion

The neural network was trained with the use of a Deep Learning Toolbox set, in the MATLAB environment. The area set contains 98 examples, with eight of them randomly selected for tests—one per each category (the “two cords missing” category does not contain any training examples and was therefore omitted). The training database consists of 90 examples. The aim of the network training process is to determine optimal weight values, which would ensure proper solutions.

This tool enables the network to self-train and also determines the validation error—when it starts to increase, the training process is interrupted so that the network does not lose its generalization capability (the ability to solve examples not encountered during the learning process) [21,22].

Using the possibilities generated by *MATLAB*, the influence of several key elements on the quality of the algorithm was investigated, and the number of neurons in the hidden layer and the type of activation function were changed. Each of the experiments was carried out 100 times in order to minimize the impact of the initial weight values on the generated solution, and the results presented in this study present average values. Table 1 contains the values obtained on successive network outputs during the testing process performed on a prepared test set of eight samples, for different numbers of hidden layer neurons $k_1 \in \{1, 3, 8, 20, 100\}$. The network output in which the computed value should be the highest is indicated in red.

In the process of learning a neural network, a number of factors influencing the completion or continuation of this process is investigated. One of them is the mean square error (MSE) course, which is determined on a separate (validation) data set. The result generated by the network on the examples from the validation set does not affect the weight matrix, but, on its basis, the MSE is determined. At some point, the validation error starts to increase, and the network loses the ability to generalize, so it is very important to interrupt the learning process in the event of such a situation and restore the value of weighting factors until the validation error is as small as possible. The learning process is also interrupted when the achieved MSE error drops below a certain expected value on the validation set. Table 2 summarizes the results of the basic parameters read in the learning process of the neural network.

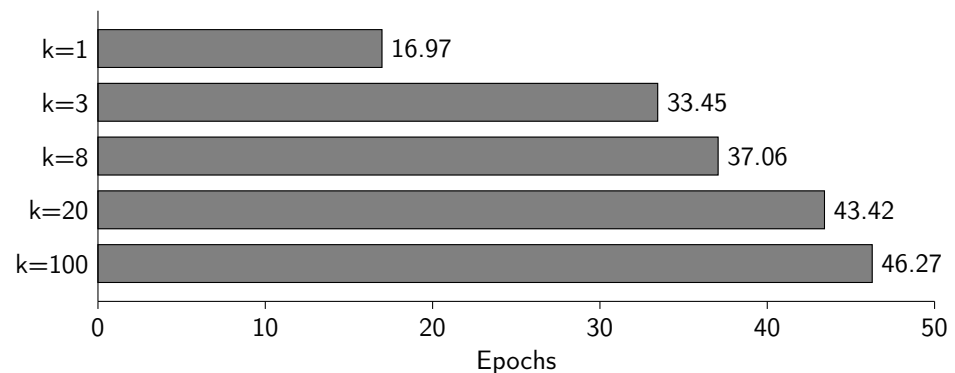
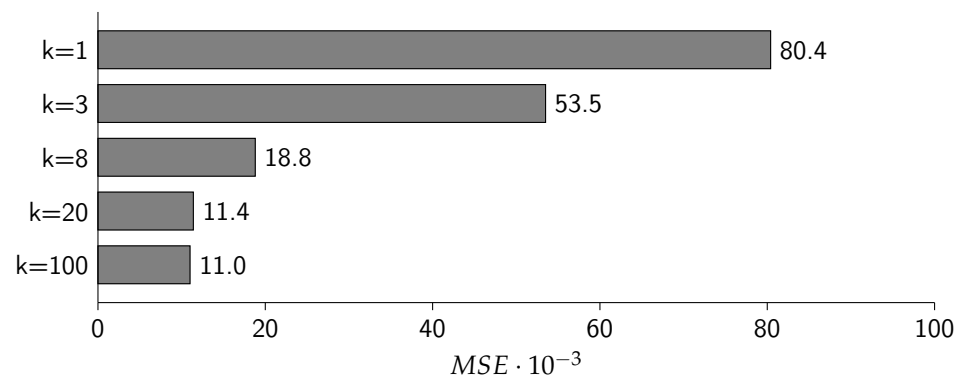
Table 1. Results obtained in network tests (impact of the number of neurons in the hidden layer).

Number of Neurons in the Hidden Layer	Network Output								
	1	2	3	4	5	6	7	8	9
$k_1 = 1$	0.26	0.00	0.07	0.07	0.14	0.08	0.16	0.16	0.05
	0.09	0.00	0.12	0.08	0.10	0.09	0.17	0.20	0.13
	0.15	0.00	0.09	0.12	0.14	0.09	0.15	0.17	0.07
	0.10	0.00	0.06	0.08	0.36	0.11	0.11	0.14	0.02
	0.06	0.00	0.09	0.09	0.23	0.13	0.14	0.18	0.05
	0.18	0.00	0.09	0.07	0.11	0.08	0.18	0.19	0.09
	0.16	0.00	0.09	0.07	0.12	0.09	0.18	0.19	0.07
	0.07	0.00	0.11	0.06	0.05	0.07	0.14	0.17	0.29
$k_1 = 3$	0.56	0.00	0.02	0.05	0.09	0.05	0.11	0.09	0.00
	−0.03	0.00	0.22	0.04	−0.03	0.09	0.23	0.31	0.13
	0.11	0.00	0.04	0.31	0.16	0.14	0.09	0.09	0.05
	0.05	0.00	−0.01	0.12	0.69	0.14	−0.01	0.00	0.01
	−0.03	0.00	0.10	0.14	0.24	0.17	0.14	0.19	0.03
	0.10	0.00	0.14	0.03	−0.04	0.08	0.32	0.33	0.01
	0.07	0.00	0.13	0.04	0.02	0.10	0.30	0.33	−0.02
	0.01	0.00	0.11	0.02	−0.01	0.02	0.03	0.05	0.69
$k_1 = 8$	0.59	0.00	−0.02	−0.02	0.15	0.06	−0.03	0.26	0.00
	0.01	0.00	0.92	0.02	0.04	−0.01	0.02	0.02	0.00
	0.04	0.00	0.01	0.97	−0.05	0.07	0.01	−0.03	0.00
	0.03	0.00	0.01	0.09	0.73	0.18	0.02	−0.05	0.00
	−0.08	0.00	−0.01	0.07	0.09	0.81	0.01	0.09	0.00
	0.02	0.00	0.09	−0.02	−0.05	0.06	0.75	0.13	0.00
	0.09	0.00	−0.09	−0.03	0.04	0.05	0.18	0.74	0.00
	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.81
$k_1 = 20$	0.71	0.00	−0.02	−0.01	0.11	0.02	0.00	0.19	0.00
	0.03	0.00	1.00	0.01	0.02	0.00	−0.08	0.05	0.00
	0.02	0.00	0.00	1.02	−0.04	0.02	0.00	−0.01	0.00
	0.02	0.00	0.00	0.06	0.76	0.18	0.02	−0.03	0.00
	−0.04	0.00	0.00	0.01	0.03	0.95	0.00	0.03	0.00
	−0.03	0.00	0.05	0.00	−0.01	0.04	0.89	0.06	0.00
	0.05	0.00	−0.07	−0.01	0.03	0.05	−0.01	0.95	0.00
	0.00	0.00	0.01	0.00	−0.01	0.01	−0.01	0.00	0.74
$k_1 = 100$	0.71	0.00	−0.02	−0.01	0.11	0.01	0.01	0.22	0.00
	0.02	0.00	1.02	0.00	0.03	0.01	−0.12	0.05	0.00
	0.01	0.00	0.00	1.03	−0.04	0.01	−0.01	0.00	0.00
	0.03	0.00	0.00	0.07	0.77	0.17	0.00	−0.03	0.00
	−0.05	0.00	0.00	0.02	0.03	0.94	0.02	0.03	0.00
	−0.03	0.00	0.05	0.02	−0.04	0.03	0.90	0.06	0.00
	0.03	0.00	−0.06	−0.01	0.06	0.05	0.03	0.90	0.00
	0.00	0.00	0.02	0.00	0.00	0.01	−0.01	0.00	0.69

Table 2. Parameters of the network.

Number of Neurons in the Hidden Layer	Epochs	MSE
$k = 1$	16.97	0.0804
$k = 3$	33.45	0.0535
$k = 8$	37.06	0.0188
$k = 20$	43.42	0.0114
$k = 100$	46.27	0.0110

The charts below show the data from Table 2 in bar charts (Figures 9 and 10).

**Figure 9.** Number of epochs until the end of the learning process.**Figure 10.** Mean square error.

A classification is considered correct if one of the network outputs assumes a value above the predefined acceptance threshold, and all of the remaining outputs assume values below the rejection threshold. With the values being 0.6 for the acceptance threshold and 0.4 for the rejection threshold, the accuracy of the network may be evaluated as follows (Table 3):

Table 3. Neural network accuracy.

Number of Neurons in the Hidden Layer	Correctly Detected	Accuracy [%]
$k = 1$	69	8.63
$k = 3$	218	27.25
$k = 8$	616	77.00
$k = 20$	656	82.00
$k = 100$	670	83.75

For the best selected number of neurons in the hidden layer ($k = 20$), tests were carried out on the impact of the selected activation function in the hidden layer of the neural network on the quality of recognition. For this purpose, four functions were used: *logsig*, *radbas*, *softmax*, *tansig*, and the test results were collected in Table 4.

Table 4. Results obtained in network tests (impact of activation function in the hidden layer).

Activation Function	Network Output								
	1	2	3	4	5	6	7	8	9
<i>logsig</i>	0.91	0.00	−0.01	0.02	0.01	−0.04	0.07	0.05	0.00
	0.00	0.00	1.02	0.01	0.04	−0.03	−0.10	0.07	0.00
	0.01	0.00	0.00	0.71	0.22	0.07	−0.01	0.00	0.00
	0.04	0.00	0.00	−0.06	0.97	0.06	−0.02	0.00	0.00
	−0.06	0.00	−0.01	0.04	0.07	0.88	0.01	0.05	0.00
	0.01	0.00	0.03	−0.01	−0.02	0.10	0.68	0.20	0.00
	0.04	0.00	−0.03	−0.02	0.04	0.04	0.23	0.65	0.00
	−0.03	0.00	−0.05	0.00	0.04	−0.03	0.07	0.13	0.80
<i>radbas</i>	0.98	0.00	0.00	0.01	0.00	−0.03	0.01	0.06	0.00
	0.00	0.00	1.03	0.01	0.01	0.01	−0.06	0.03	0.00
	−0.01	0.00	0.01	0.74	0.21	0.05	0.00	−0.01	0.00
	0.02	0.00	0.01	−0.04	0.98	0.03	−0.02	0.00	0.00
	−0.02	0.00	0.00	0.01	0.00	0.98	0.00	0.01	0.00
	−0.04	0.00	0.01	−0.01	−0.01	0.14	0.78	0.12	0.00
	0.00	0.00	−0.02	−0.01	0.02	0.04	0.23	0.68	0.00
	0.08	0.00	0.00	0.12	0.17	−0.10	−0.06	0.10	0.61
<i>softmax</i>	0.91	0.00	0.01	0.02	0.00	−0.04	0.10	0.00	0.00
	0.01	0.00	0.89	0.01	0.03	0.00	0.00	0.05	−0.01
	0.03	0.00	0.01	0.54	0.22	0.21	0.02	−0.03	0.00
	0.04	0.00	0.01	−0.07	0.92	0.13	−0.02	−0.03	0.00
	−0.08	0.00	−0.02	0.22	0.20	0.47	0.03	0.17	0.00
	−0.02	0.00	0.11	−0.01	−0.02	0.05	0.55	0.32	0.00
	0.07	0.00	−0.05	−0.02	0.02	0.08	0.33	0.54	0.00
	−0.01	0.00	−0.06	0.00	0.00	0.00	0.02	0.03	0.73
<i>tansig</i>	0.95	0.00	−0.01	0.02	0.00	−0.02	0.05	0.04	0.00
	0.02	0.00	1.03	0.00	0.03	0.01	−0.09	0.04	0.00
	0.01	0.00	0.00	0.72	0.24	0.05	−0.01	0.00	0.00
	0.02	0.00	0.00	−0.05	0.98	0.04	−0.03	0.00	0.00
	−0.04	0.00	0.00	0.02	0.03	0.94	0.01	0.03	0.00
	0.00	0.00	0.02	−0.01	−0.01	0.14	0.70	0.16	0.00
	0.03	0.00	−0.03	−0.01	0.03	0.04	0.24	0.66	0.00
	0.04	0.00	−0.02	0.02	0.01	−0.01	0.07	0.04	0.63

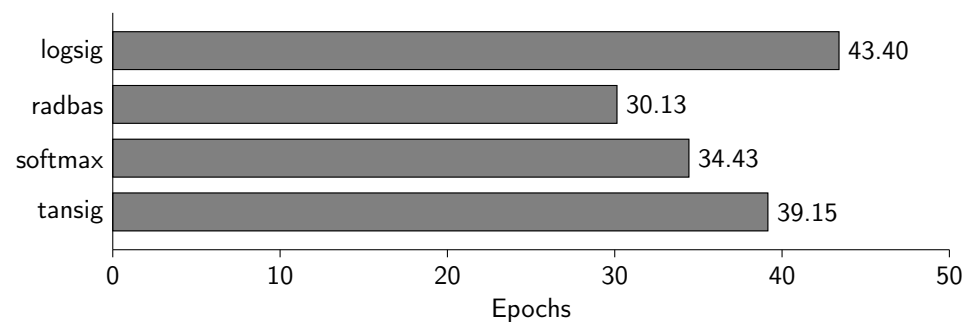
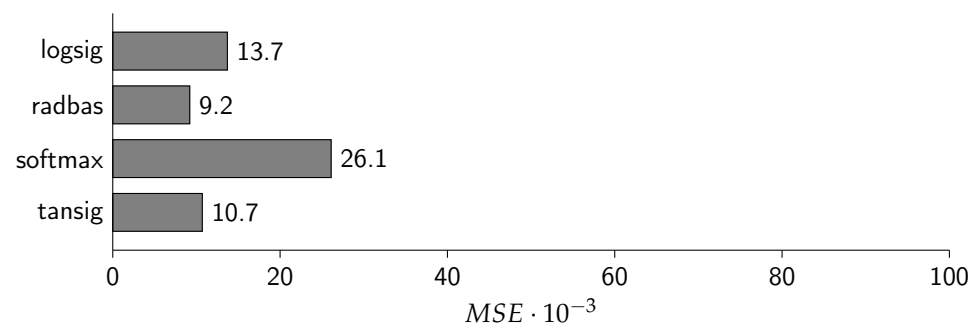
As before, in the learning process of the neural network, the MSE indicator was observed, and the learning process was interrupted when the MSE value on the validation set began to increase (the patience parameter was set to 6). Table 5 contains averaged parameters obtained in the network training process.

A classification is considered correct if one of the network outputs assumes a value above the predefined acceptance threshold, and all of the remaining outputs assume values below the rejection threshold, with the values being 0.6 for the acceptance threshold and 0.4 for the rejection threshold. Table 6 collects the network performance data with the activation function selected.

Table 5. Parameters of the network.

Activation Function	Epochs	MSE
<i>logsig</i>	43.40	0.0137
<i>radbas</i>	30.13	0.0092
<i>softmax</i>	34.43	0.0261
<i>tansig</i>	39.15	0.0107

The charts below show the data from Table 5 in bar charts (Figures 11 and 12).

**Figure 11.** Number of epochs until the end of the learning process.**Figure 12.** Mean square error.**Table 6.** Neural network accuracy.

Activation Function	Correctly Detected	Accuracy [%]
<i>logsig</i>	647	80.88
<i>radbas</i>	648	81.00
<i>softmax</i>	506	63.25
<i>tansig</i>	660	82.50

4. Conclusions

The popularity of artificial neural networks as tools for solving classification problems results mainly from the fact that such a network can often make more accurate decisions than a team of experts, and in addition it can be implemented in almost any branch of science. A neural network designed for medical applications by researchers from the U.S. and from China achieves correct patient diagnosis rates often higher than teams of medical experts. For example, correct diagnosis rate for asthma is 90%, while the rate achieved by doctors is within 82–90% [23].

Data from magnetic sensors are well suited for further processing in search for potential damage, and the two-dimensional record mode of the data facilitates the processing and subsequent verification of the processed data. A correct manual recognition of the potential

problems requires a well-trained operator, who identifies the most characteristic points allowing an area to be classified into a particular category. Therefore, application of artificial neural networks for solving such problems may prove advantageous, regardless of the need to search for values which could serve as characteristics for a given category. This is because the network will, during the learning process, determine the input weights in order to evaluate which element of the input sequence is more significant in a given category.

The application of artificial intelligence in the form of neural networks to the classification of damage to the conveyor belt seems a promising research direction. The preliminary research results described in this article confirm that the potential of artificial neural networks can be used also in this area. In order to obtain a higher ability of the network to classify damaged areas, adding further input values should be considered (e.g., a shift of the center of gravity of a particular sub-area from the geometric center, modification of the distances between the centers of gravity of individual sub-areas, or even changes of the area span in both directions).

The improvement of the results obtained by the designed neural network would certainly be obtained with the introduction of more training data. The use of the increasingly popular deep neural networks would also be worth considering. The development of these networks could be possible by the advancement of computer techniques, including increasing the computing power of computers, parallel data processing, and also thanks to the development of fast computing algorithms. Deep neural networks are able, somehow automatically, to extract diagnostic features for a given class of objects without supervision, thus they excuse the user from the difficult problem of defining and then selecting the key diagnostic features of the examined process [17].

It is worth noting that the selection of neural network parameters has a significant impact on the quality of its operation. As part of the research described here, both the effect of the number of neurons in the hidden layer of the network and the type of activation function used in it were checked. It can be seen that the network achieves the best results for 20 (or 100) neurons in the hidden layer—both the recognition efficiency is the highest, amounting to 82.00 % (or 83.75 %), and the MSE error is the smallest—0.0114 (or 0.0110). A network with more neurons in the hidden layer learns over more epochs, and each epoch takes longer (because more operations have to be performed). It can also be emphasized that the testing process consists only of a one-time multiplication of the input signal by the matrix of selected weights, and therefore the time needed to carry out the testing process is imperceptible; therefore, the once learned network can recognize patterns without the need to guarantee its high computing power. As part of this study, both the input database and the input signal data size are small, and thus the learning time is imperceptible; however, in the case of large data sets or feeding a larger data vector into the input, the operating time can be significantly extended. The solution to problems with long network training time is to use a graphics processor, which is adapted to perform parallel calculations, and thus guarantees the acceleration of numerical calculations performed on it.

One factor that allows you to tune your neural network and make it more efficient is the use of various activation functions. As part of the study described here, the results resulting from the use of various functions are similar, slightly exceeding the effectiveness of 80%, except for the result related to the *softmax* function—here the recognition efficiency decreased significantly and the MSE error increased. It can be emphasized that the best possible set of parameters for the studied number of networks is a network with 20 neurons and the activation function *radbas* or *tansig* in the hidden layer. The use of the radial function presents the smallest MSE error value and stops on average after the lowest number of epochs (30.13); however, the recognition efficiency of the radial layer network is 81.00%, and, when the *tansin* function is used, the recognition efficiency slightly increases to 82.50%.

Author Contributions: Conceptualization, A.K.-B. and A.R.; methodology, A.R.; validation, A.K.-B.; formal analysis, A.K.-B. and A.R.; resources, A.K.-B.; data curation, A.R.; writing—original draft preparation, A.K.-B., and A.R.; writing—review and editing, A.R.; visualization, A.R.; supervision, A.K.-B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received funding from grant LIDER/41/0227/L-10/18/NCBR/2019.

Data Availability Statement: The data in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Webb, C.; Sikorska, J.; Khan, R.N.; Hodkiewicz, M. Developing and evaluating predictive conveyor belt wear models. *Data-Cent. Eng.* **2020**, *1*, e3. [CrossRef]
- Żur, T.; Hardygóra, M. *Belt Conveyors in Mining*; Śląsk: Katowice, Poland, 1996.
- Błażej, R. *Assessment of the Technical Condition of Conveyor Belts with Steel Cords*; Oficyna Wydawnicza Politechniki Wrocławskiej: Wrocław, Poland, 2018.
- Neumann, T.; Conveyor Belt Group. Mining 4.0—Our Digital Journey. In Proceedings of the Smart Mining Conference Forum Bergbau 4.0, Aachen, Germany, 14–15 November 2017.
- Kirjanów-Błażej, A.; Jurdziak, L.; Błażej, R.; Kozłowski, T. Identification of the Development of Failures to the Core of Steel Cord Conveyor Belts in Space and Time. Part 1—Failures Distribution along the Axis of the Belt, Wrocław 2019. Available online: https://www.researchgate.net/publication/333668369_Identification_of_the_development_of_failures_to_the_core_of_steel_cord_conveyor_belts_in_space_and_time_Part_1_-_failures_distribution_along_the_axis_of_the_belt (accessed on 12 June 2021).
- Błażej, R.; Jurdziak, L.; Kirjanów, A.; Kozłowski, T. Core damages increase assessment in the conveyor belt with steel cords. *Diagnostyka* **2017**, *18*, 93–98.
- Błażej, R.; Jurdziak, L.; Kozłowski, T.; Kirjanów, A. The use of magnetic sensors in monitoring the condition of the core in steel cord conveyor belts—Tests of the measuring probe and the design of the DiagBelt system. *Measurement* **2018**, *123*, 48–53. [CrossRef]
- Diagbelt-PL. Available online: <http://www.diagbelt.pwr.edu.pl/index.php/pl/> (accessed on 26 February 2021).
- Kirjanów, A. Model of the Development of Core Damage in Conveyor Belts with Steel Cords. Ph.D. Thesis, Wrocław University of Science and Technology, Wrocław, Poland, 2018.
- Olchówka, D.; Rzeszowska, A.; Jurdziak, L.; Błażej, R. Statistical Analysis and Neural Network in Detecting Steel Cord Failures in Conveyor Belts. *Energies* **2021**, *14*, 3081. [CrossRef]
- Selection of Measurement Parameters Using the DiagBelt Magnetic System on the Test Conveyor. Master's Thesis, Wrocław University of Science and Technology, Wrocław, Poland, 2020.
- Jurdziak, L.; Błażej, R.; Burduk, A.; Bajda, M.; Kirjanów-Błażej, A.; Kozłowski, T.; Olchówka, D. Optimization of the operating costs of conveyor belts in a mine in various strategies of their replacement and diagnostics. *Transp. Przem. Masz. Rob.* **2020**, *4*, 14–22.
- Tadeusiewicz, R. *Neural Networks*; Akademicka Oficyna Wydawnicza: Warszawa, Poland, 1993.
- Burduk, A.; Więcek, D.; Zajačko, I.; Jurdziak, L.; Błażej, R. Analysis and risk assessment of the supply of copper ore in the belt conveyor system in an underground mine. *Acta Montan. Slovaca* **2020**, *25*, 236–249.
- Więcek, D.; Burduk, A.; Kuric, I. The use of ANN in improving efficiency and ensuring the stability of the copper ore mining process. *Acta Montan. Slovaca* **2019**, *24*, 1–14.
- Osowski, S. *Neural Networks in Algorithmic Use*; WNT: Warszawa, Poland, 1996. (In Polish)
- Osowski, S. *Neural Networks for Information Processing*; Oficyna Wydawnicza Politechniki Warszawskiej: Warszawa, Poland, 2020.
- Floreen, P. The convergence of Hamming memory networks. *IEEE Trans. Neural Netw.* **1991**, *2*, 355–365. [CrossRef] [PubMed]
- Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* **1989**, *2*, 359–366. [CrossRef]
- Kołmogorov, A.N. On the representation of continuous functions of many variables by superposition of continuous functions of one variable in addition. *Dokl. Akad. Nauk ZSRR*, **1957**, *114*, 953–956.
- Demuth, H.; Beale, M. *Neural Network Toolbox User's Guide*; MathWorks, Inc.: Natick, MA, USA, 2014.
- Anghel, D.; Ene, A.; Belu, N. A Matlab Neural Network Application for the Study of Working Conditions. 2013. Available online: <https://www.scientific.net/AMR.837.310> (accessed on 12 June 2021).
- Metz, C. In China, Training A.I. To Diagnose Diseases. *New York Times*, 12 February 2019.