

Article

Idea of Using Blockchain Technique for Choosing the Best Configuration of Weights in Neural Networks

Alicja Winnicka and Karolina Kęsik*

Institute of Mathematics, Silesian University of Technology, Kaszubska 23, 44-100 Gliwice, Poland

* Correspondence: Karola.Ksk@gmail.com

Received: 4 July 2019; Accepted: 6 August 2019; Published: 10 August 2019



Abstract: The blockchain technique is becoming more and more popular due to its advantages such as stability and dispersed nature. This is an idea based on blockchain activity paradigms. Another important field is machine learning, which is increasingly used in practice. Unfortunately, the training or overtraining artificial neural networks is very time-consuming and requires high computing power. In this paper, we proposed using a blockchain technique to train neural networks. This type of activity is important due to the possible search for initial weights in the network, which affect faster training, due to gradient decrease. We performed the tests with much heavier calculations to indicate that such an action is possible. However, this type of solution can also be used for less demanding calculations, i.e., only a few iterations of training and finding a better configuration of initial weights.

Keywords: algorithms; blockchain; machine learning

1. Introduction

The implementation of 5G technology is getting closer and closer. In selected cities around the world, action tests are already being carried out in terms of speed or the number of connected devices in a specific area. The selection of parameters during testing is important because thanks to them, it is possible to create a real Internet of Things. The word *real* is important in the context of connecting and allowing sending data between a huge number of devices. This is especially important for intelligent cars, which should be able to exchange information with passing objects, e.g., lanterns. However, it is easier to write than to do. The reason for this is the amount of transferred data (in the case of images from cars cameras, the amount of data can be huge) or the number of objects wanting to send data in a small area (an example is a traffic jam). Using the 5th generation mobile technology network, both problems should be solved.

It is worth noting that the amount of data transmitted even in 5G network can be very burdensome for servers. Parallel recording of very large amounts of data can cause discs to jam. In addition, it will need enormous computing power to process this information. This is particularly important in the case of using artificial intelligence techniques, which learning process depends on the quality and amount of samples. And it can last up to several months. And practical application forces constant training of such solutions to enable their development because more and new data are downloaded. Another problem is overfitting. Imagine a situation that data flows to the server all the time for a few days. At this time, they will be processed, and then the classifier will be trained with them. After a long period, the required effectiveness will be achieved, but there might be a problem of overtraining. As a result, we could have automatic decisions that should not necessarily take place and may endanger people's lives.

The solutions to such problems can be found under the name *edge artificial intelligence*, where it is understood primarily as performing the required operations on the local equipment, and not somewhere outside, an example of which is the server. The use of this type of technology ensures data

security, as they are not distributed anywhere else than on their own. In addition, data processing can be performed in the background, regardless of other actions. Consequently, additional elements, such as training, can be repeated using more and more new data in the background with a predefined load of hardware power. This is important because of the basic operation of the device. This is especially important on the Internet of Things where the user can use one of the devices. An example is a smartwatch that records pressure, body temperature or even the outside temperature. This data can be very different if the environment is changed. The use of artificial intelligence in this case may require overtraining of the used classifier to best adapt to the user's needs. And this can be done locally, while continuing to use the device. An additional advantage will be the ability to do this without using the Internet, because there are situations when the user has no access to it. Of course, such a solution brings with it some disadvantages. The greatest of them is the computing power on given devices, because the application of artificial intelligence will require it. This problem can be solved by various methods. From the use of better technical components to implementation solutions.

The mentioned implementation issue is a more interesting solution that does not require additional financial expenses. However, this requires the creation of supervisory mechanisms that enable parallel execution of two threads, as well communication and possible exchange of information between user's devices in the local network. As a local network, we understand several devices that can work together and belong to a single user. Such action should guarantee protection of the data being processed as well as increase of functionality and its effectiveness. The reason is to increase the possibility of making decisions based on various data. In this work, we propose a solution based on the idea of blockchain. More precisely, increasing the flexibility of objects on the Internet of Things using the idea of concluding transactions. The technique described above is intended for artificial intelligence methods, i.e., re-training classifiers in the event of an urgent need. Such a need may be attaching a new device or major changes in the user's functioning.

2. Related Works

Edge artificial intelligence finds more and more possibilities for practical use. This is visible not only in the industry, where we find information about implemented intelligent technologies aimed at improving and simplifying our lives but also in the work carried out by scientists. Thanks to which it is possible to use it in industry. Scientific and development work focuses almost on every aspect of artificial intelligence and its action on the Internet of Things.

The artificial intelligence itself is developing in terms of newer models. The purpose of this is primarily to create self-learning algorithms. In [1], the authors looked at the latest achievement in this field, and in particular they noticed big steps in image/sound recognition. In this paper, the authors presented two techniques for explaining predictions of deep learning models. The effect of their work was to indicate tools for detecting defects in such models, which is important for their further improvement. It was noted that, in 2006, artificial intelligence defeated the world champion in GO, a game much more complicated than chess [2]. Such a big step forward in this field, allowed much more real use of these techniques in other areas of our lives. This is evident in the analysis of big data that is produced at almost every step. Even the most ordinary smartphone downloads data and can use them for various purposes, hence it is important to correctly classify them [3,4]. One of the most desirable applications is the rapid detection of diseases that are treated as non-curable, more specifically, cancers. Only in the initial phase of the disease, they can be cured. The problem is that it is almost impossible to detect them at this stage. Therefore, artificial intelligence is used to search medical data, often in the form of images, and search for the first cancer cells [5,6].

Various applications of artificial intelligence and better and better results make it an urgent area of computer science. As already noted, high hopes are also associated with the Internet of Things, where all of these applications can be used. However, there are some limitations in computing power if data operations are performed on local devices connected to the network. Hence the problem of network economics and resource allocation is very important. There are many different solution

to this problem and two of them are described in [7], where the idea of the auction is shown in order to perform certain activities. These solutions also need some implementation architectures, because all these devices have to be somehow connected to each other [8]. An interesting solution to this implementation is the mechanism called *fog computation* [9] and it is an extension of operations in the clouds to the edge of the enterprise network. Of course, scientists from around the world are working on different concepts, for instance in [10], the authors described idea to minimize the edge resources but satisfying QoS requirements. On the other hand, they tried to solve problems related to scalability, security, resilience and real-time data delivery in cloud environment. In [11] was proposed blockchain-based model of distributed cloud architecture, which allowed to reduce the costs of high-performance computing.

In addition, blockchain technology and cryptocurrency mining are ubiquitous in the diverse trends of science and industry [12,13]. Blockchain is mainly found in large networks of interconnected devices [14–16]. What is visible in creating smart cities, where every user uses energy resources. Such resources can be collected, analyzed and sent between the supplier and clients. Everything is based on the principle of using a given resource unit, after which another one is created, and thus another block is created. The impact of the technique is interesting because the information exchange works on a peer-to-peer basis and this data is well protected. Although the idea is based on asymmetric cryptography, it is still developing the improvement of data closure due to undesirable access [17,18]. Of course, this technology has its limitations, hence it is important to focus on the structure itself and its optimization [19]. An important element is also to focus on the good of the platform for the use of blockchain. In [20,21], the authors presented different platforms and their advantages and drawbacks.

From the presented state of knowledge and recent achievements in artificial intelligence, Internet of Things and blockchain, it can be seen that these are trends developed due to industrial needs. Therefore, it is important to focus also on decentralized databases and enable local devices to deal with private data, so that every user (not only private ones but whole companies) has a sense of security and continuous improvement of their own devices without controlling them.

In this paper, we present an blockchain technique for the use in edge AI, i.e., an algorithm for finding better configurations of weights in neural networks using the blockchain idea. The idea is based on the general mechanism of finding neural network's weights in a group of specific devices. This paper is composed of six sections, where the first introduces the topic AI and blockchain. Then, deep neural networks and its mathematical model are introduced. In the next section, an algorithm based on the blockchain idea that could be used for edge AI is proposed. The next chapter shows the effects of the experiments carried out and indicates the use of such solutions in practice. In the last section, the proposed solution and results are summarized.

3. Deep Neural Networks

The history of neural networks dates back to the early 1940s, when McCulloch and Pitts published the first mathematical model of the nerve cell's operation [22]. In a short time, networks began to be constructed, i.e., structures consisting of many neurons, where neurons are arranged in so-called layers (or columns). Neurons, in particular layers were connected with each other by means of bonds simulating synapses in human brain. Each connection of this type was burnt with a random weight. The weight values are modified to adapt the network to perform a specific problem.

Mentioned neuron has a simply construction. As already noted, neuron is connected to each other in relation to the layers, what allows to deduce the general idea of operation. Each unit will receive some data from the neurons in the previous layer and send them to the next layer. The incoming information will contain the value of the neuron x_i and the weight w_i on the synapse through which this data is transmitted. The received information are processed by calculating the sum of the product of these weights and the information. Then, this value is scaled by the activation function $f(\cdot)$ and sent further. It can be illustrated by the following equation

$$y_i = f(e) = f\left(\sum_{i=0}^n x_i w_i\right), \quad (1)$$

where n is the number of neurons in previous layer. In the case of activation function, the most common one is hyperbolic tangent, but the selection is made in relation to expected output. For instance, one of such a function is hyperbolic tangent, which range is in $(-1, 1)$ and the mathematical formulation of this is

$$f(x) = \tanh(x) = \frac{2}{1 + \exp(-2x)} - 1. \quad (2)$$

General idea of neural network is based on the construction the grid of neurons composed as layers. There are three types of layer. The first one is called input, and it takes data and process them further. The next type is called hidden and combine intermediate layers. Their number may be different, in contrast to other types. The last one is known as output one, which return the results of classification.

Such a structure without learning process is useless. One of the most known is called back-propagation algorithm [23] and it is a basic model of training for the proposed solution. Algorithm works on the principal of minimization of errors in whole structure, which is based on gradient descent. To present the idea of this algorithm. Let us assume, that the set of samples consists numerical values inserted into a vector and described as $\{x_1, x_2, \dots, x_n\}$. This set is divided into two subsets in proportion of 70% to 30% of all samples. The first set is understood as training, and the second as validation one. Using training data, algorithm will modify the weights in whole network. Additionally, the synapses will be marked as mn (synapse that connects neuron m with n), so the weight on this connection can be labeled as w_{mn} . At the beginning of creating this structure, all weights are selected in random way in the range of $(0, 1)$. In each iteration, when one training vector is given at the input layer, and this data are processed by all layers, an error of network is calculated. Error is defined as difference between expected value z and obtained y , so it can be presented in following way

$$\delta = z - y. \quad (3)$$

Processing training vectors through the network from input to output layers is understood as a step forward. Training works on the principle of modifying weights from output to input, so it can be called step back. Calculated error in the end of forward processing must be processed back to all neurons and it can be formulated as

$$\delta_i = \sum_k w_{ik} \delta_k, \quad (4)$$

where k is the number of neurons in the next layer for the i -th neuron. Using this errors, the weights can be recalculated as

$$w'_{nm} = w_{nm} + \xi \delta_m \frac{df(e)}{de} y_n, \quad (5)$$

where ξ is a coefficient influencing the speed training and this value should be decreased in each iteration.

4. Blockchain Strategy for Edge Artificial Intelligence

The blockchain strategy, as the name indicates, is a list of blocks denoting records that we use to store and transmit certain transaction data. Basic structure in this strategy is block, which is a virtual objects composed of stored data—by using its number and time of registration of data. In addition, each block has its own digital signature called hash. This signature refers to the previous block and is included in the block of data. Thanks to this hash, the blocks retain the structure of the chain, because those blocks are connected to each other through a given hash. In the original blockchain operation, this hash is a 256 bit number that is created by the function called SHA-256 (*Secure Hash Algorithm*). Blockchains are also known as a huge distributed public ledger made of

existing transactions. The main process, which is called mining, is to add another transaction to blockchain. In every process of mining participate miners—single computers or groups of computers, whose goal is to get cryptocurrency. Thanks to this algorithm, all miners (or users) can compete with each other in order to get a certain, specific reward for completing the transaction. In general, this is how blockchain works.

The foundation of the blockchain, concept of the Proof of Work (PoW), was developed to increase security against various attacks. The idea requires from any user some work to be done—generally by using computer calculations, which allows further access. After several years, Bitcoin presented a novel way to use this idea. This use of PoW idea involves the validation of transaction—and after that it can be added to blockchain. Miners are competing with others to solve puzzle, which is very difficult to be solved, but fairly simple to be verified very fast. If any miner solves a problem, he makes his solution available and its correctness is verified by other miners. If confirmed, the block is attached to the blockchain and the miner receives a reward in the form of cryptocurrencies.

The main advantages of this technique is the basic peer-to-peer principle with no central servers. Each device in a this network structure has the same rights to send information or verify them. However, each device has the ability to view only its own transactions. The described advantages make that, in this technique, information are available for all devices, but simultaneously are protected against unauthorized access. This type of technique can be used on the Internet of Things, where we have many devices that connect with each other. Let us illustrate the main components of the proposed strategy. This strategy is only inspired by the blockchain, because the principles of digging the currency are rewritten from scratch, and the structure itself used as a carrier.

Knowing the general idea, we can focus on the presentation of another, more practical solution due to the Internet of Things and artificial intelligence. Each demand is included as a transaction. One component sends a demand for a certain task, an example of which can be training the classifier using selected data for a specific error. Such a task is waiting for the miners to submit their application. We assume that the maximum number of applicants is pre-determined as n . Next, the task is assigned and handled until it is executed or conditionally stopped. Conditional stopping will be called extortion when other tasks with higher priority appear in the queue. In the event of such a situation, the miners who performed this task compare their results and better write to the block as a completed task. And they take the next in order not to cause too long stagnation—this is due to the construction of the original blockchain idea, because there is a rule that adding another block is depending on the end of the previous one.

However, it is worth noting that the very idea of recording the best result (though not required) is not always profitable. This is particularly visible if the results of a given task have an impact on the next one. Therefore, a rule should be added to require another broadcast of the same transaction in the future. In the case of the use of artificial intelligence methods in this construction, more precisely classifiers, their training would be performed again to obtain a stop condition. It should be noted that such constant improvement is not a bad solution—in the absence of new demands such as software upgrade or adding another device to the network, continuous improvement results in better efficiency.

It was mentioned earlier that every transaction is done for a reward. Each task has its priority and we make the reward dependent on it. When the priority is higher, the rewards will be greater. As a reward, we mean a virtual currency denoting hardware usage. The more you earn for a particular device, the more it is known that the equipment is more used. And hence, there is a much greater likelihood of its consumption and replacement. This type of solution will allow control over the operation of the equipment and predict where the problem may occur in the future. To make this scheme more realistic, let us note the fact that a given task can be performed by n devices. In addition, it is possible to interrupt a task when another one with a higher priority appears. Taking into account such action, reward each device performing the task to maintain the real state of equipment wear. Therefore, the proposed scheme can be illustrated as follows

$$\begin{cases} 0.5 \omega & \text{for completing the task,} \\ \frac{0.5 \phi \omega}{n} & \text{for trying to complete the task,} \end{cases} \quad (6)$$

where ϕ is the percentage value of the completed task, ω is the maximum value of the reward ϕ multiplied by priority ζ in the following way

$$\omega = \zeta \phi. \quad (7)$$

It is easy to see that the amount of a given currency in the network is infinite and accumulates indicating hardware consumption.

After completing the task, before the miner is rewarded, a verification process takes place, where each member of the network verifies the transferred data. Verification is performed by downloading the given data and testing for their own data. In the case when the efficiency is less than $\alpha\% \zeta$ (minimum threshold, which is increased depending on the priority), miners get their reward, but the task itself has an increased priority of one. The interruption occurs when there is another task, the one that has the higher priority at the moment is selected. Described idea is depicted on Figure 1, where data about configuration and weights are stored on-chain, and the best configuration, information about precision, and data are stored off-chain.

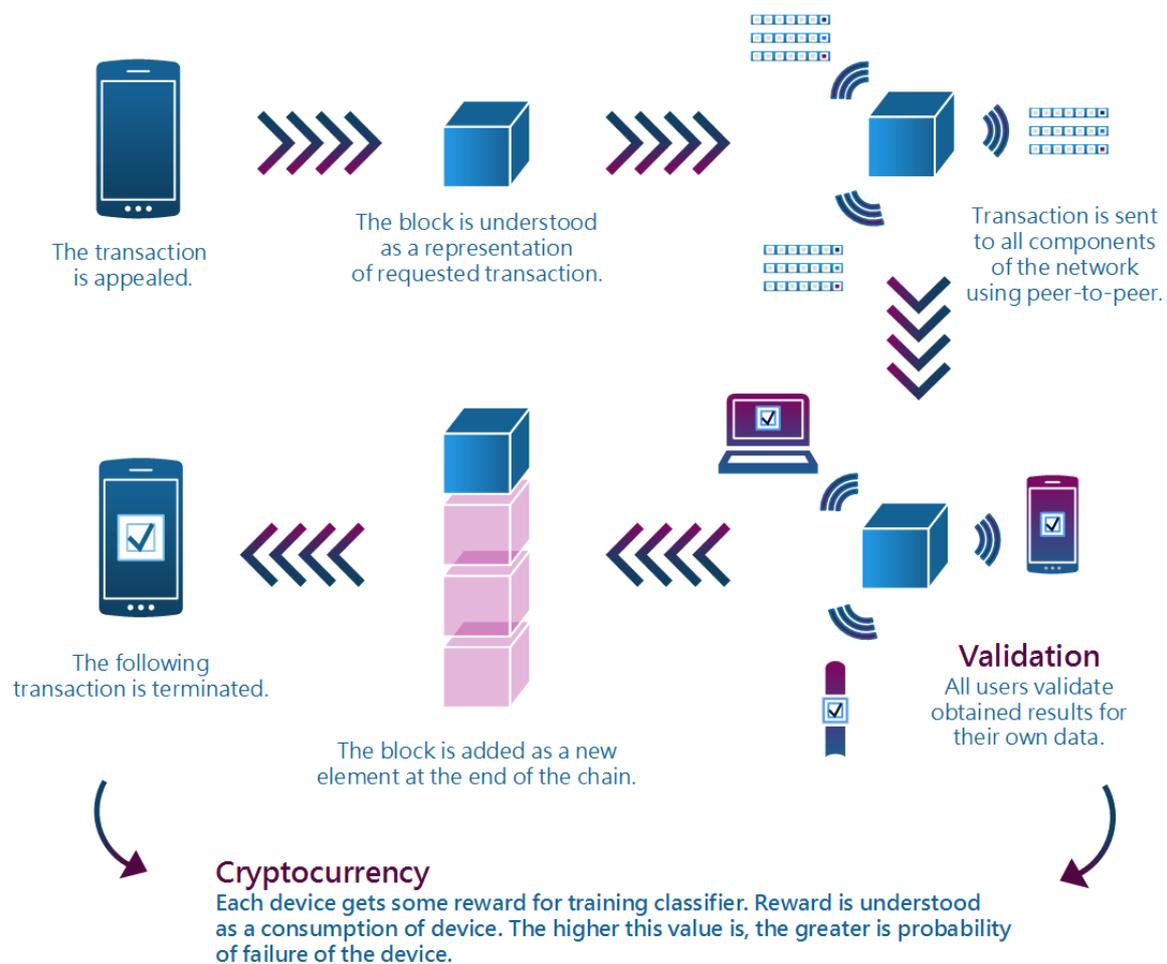


Figure 1. Visualization of the proposed blockchain-inspired technique for the operation of artificial intelligence on local environments.

5. Simulation and Experimental Evaluation

The proposed technique of training classifiers based on the idea of blockchain is a quite flexible solution in the field of artificial intelligence. However, the solution was tested due to various factors. First of all, several different devices were used and a local network was created so that they could cooperate with each other. The network was built of two sets composed of two smartphones and one smartwatch (collected data from smartwatch was transmitted to one of the smartphone). A simple application running in the background was uploaded on each device. Each program was to classify a different type of data (all saved in numerical values) independently, and then try to combine data types. Of course, in order for any functioning to be possible, the initial configuration of classifiers has been uploaded before the activity started (error of all classifiers was set to 0.1).

A list of tasks was created, i.e., creating classifiers for two types of data, then three, and finally one large classifier. The wanted error was set according to the these values {0.01, 0.001, 0.0001}. The priorities have been added appropriately as {4, 3, 2, 1}. The device had the following data obtained from the built-in sensors

- atmospheric pressure, magnetic field, temperature, humidity (smartphone),
- pulse, height, regeneration status based on 24-hour activity (smartwatch).

For each device, the classifier was trained in the simplest way, i.e., correct/incorrect user's well-being. In order for the solution to work in general, these two sets belonged to a pair of two people and formed a common network of six devices. This was done five times for five different pairs. Of course, the results were repeated for different neural network configurations—it was tested for 2, 3 and 4 hidden layers. The number of neurons in each of them was calculated as \sqrt{rt} , where r means the number of neurons in the input and t in the output layer.

Tables 1–3 shown the obtained results in terms of awards for completing task in a small network of two real devices. Users used their devices normally and in the background, neural networks training operations were performed. After training, there was a verification process where prizes were awarded. It was noticed that in the case of smaller structures, usually the operations on the smartphone were more rewarded. However, for the training process for a network composed of 4 hidden layers, the smartwatch received a larger amount. This is more visible in the charts (Figures 2–4). This may be due to the amount of work performed on the device. Because the smartwatch works with approximately equal computing power, and the smathphone depends on the user and his activity.

It is important to note that the created networks were very small and there was not much competition. Only the idea itself and checking the reward to know how much a given network component is used. The more often devices will be used, the more likely it will be to replace them in the future. The same tasks, i.e., training the network for various errors, were given in a set of 20 devices (10 smartphones and 10 smartwatch). In addition, it was assumed that a maximum of three devices can perform the task. In the proposed structure, usually three devices were calculating until the next task appeared (another task were forced to create a critical situation). The interruption appeared, but it cost additional computing power. The reason is that the task with a lower priority (and better accurate efficiency of the classifier) simply restarted the obtained results. And the process of training the classifier from an earlier task was repeated from the beginning on other devices.

The big advantage of this network model was the use of the same devices, which did not change the network configuration and re-merge the data into larger vectors. To show the sense of operation of such a solution, a server was created to which data from all devices was sent and there the classifier was trained. After the operation, the weights were sent to all devices. This solution is a classic approach to this type of problems. On the one hand, this solution does not load devices, only the server, but it forces access to the Internet and worsens the state of privacy when it comes to data. The server classifier can be trained with data regardless of the users of the product, or with data obtained from them. Both solutions are not the best ones, because the first may not be best suited to selected users, and the other one sends and saves private data outside the devices.

Table 1. Obtained results for a network composed of 2 hidden layers.

Device	Error	Priority	A Pair I		A Pair II		A Pair III		A Pair IV		A Pair V		Average	
			Reward	Time [h]	Reward	Time [h]	Reward	Time [h]	Reward	Time [h]	Reward	Time [h]	Reward	Time [h]
Smartphone	0.01	3	150	1	74	1	73	1	150	1	74	1	87	1
Smartwatch	0.01	3	71	5	150	6	150	4	72	1	150	1	98	1
Smartphone	0.001	2	48	5	100	6	100	4	100	7	49	6	66	6
Smartwatch	0.001	2	100	5	45	6	44	4	48	7	100	6	56	6
Smartphone	0.0001	1	22	12	50	14	50	13	24	16	24	18	28	15
Smartwatch	0.0001	1	50	12	19	14	23	13	50	16	50	18	32	15

Table 2. Obtained results for a network composed of 3 hidden layers.

Device	Error	Priority	A Pair I		A Pair II		A Pair III		A Pair IV		A Pair V		Average	
			Reward	Time [h]	Reward	Time [h]	Reward	Time [h]	Reward	Time [h]	Reward	Time [h]	Reward	Time [h]
Smartphone	0.01	3	150	9	150	7	66	11	64	9	68	6	83	8
Smartwatch	0.01	3	55	27	58	27	150	33	150	9	150	6	93	8
Smartphone	0.001	2	100	27	45	27	37	33	39	36	100	34	54	31
Smartwatch	0.001	2	26	27	100	27	44	33	100	36	42	34	61	31
Smartphone	0.0001	1	50	45	50	42	50	55	50	49	50	57	42	50
Smartwatch	0.0001	1	20	45	18	42	23	55	21	49	23	57	17	50

Table 3. Obtained results for a network composed of 4 hidden layers.

Device	Error	Priority	A Pair I		A Pair II		A Pair III		A Pair IV		A Pair V		Average	
			Reward	Time [h]	Reward	Time [h]	Reward	Time [h]	Reward	Time [h]	Reward	Time [h]	Reward	Time [h]
Smartphone	0.01	3	150	11	150	14	150	16	150	23	150	12	125	15
Smartwatch	0.01	3	45	45	54	14	39	16	57	23	61	12	42	15
Smartphone	0.001	2	100	45	48	42	100	44	100	55	43	49	65	47
Smartwatch	0.001	2	42	45	100	42	34	44	45	55	100	49	53	47
Smartphone	0.0001	1	50	123	50	117	50	135	50	137	23	145	37	131
Smartwatch	0.0001	1	18	123	22	117	12	135	15	137	50	145	19	131

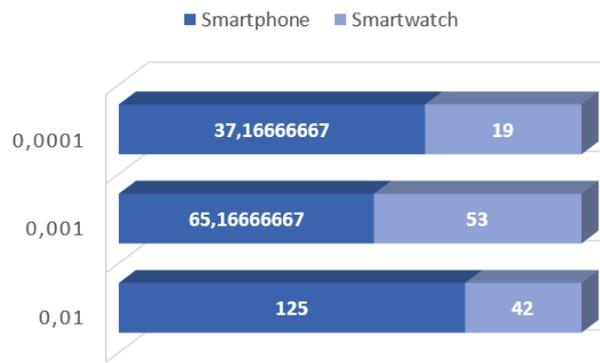


Figure 2. The value of prizes for the work (2 hidden layers) done in the proposed methodology.

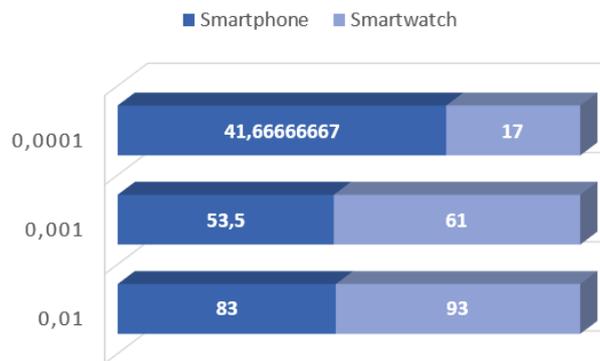


Figure 3. The value of prizes for the work (3 hidden layers) done in the proposed methodology.

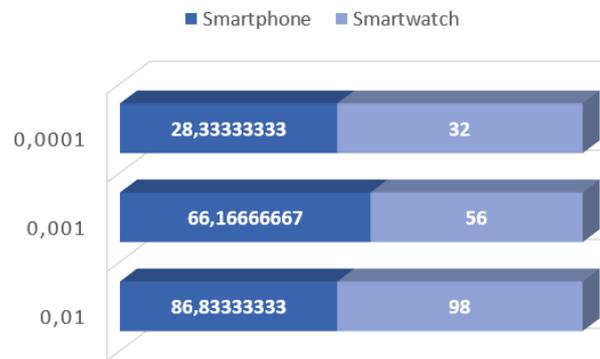


Figure 4. The value of prizes for the work (4 hidden layers) done in the proposed methodology.

Despite all the disadvantages, we tested this solution to compare it with the one described in this paper. In Figure 5, there is a graph of time depending to error. It is easy to see that using only a specific data in training process without adding more and more during this process is better to choose a classic approach to solve this problem, but when we have data that are modified during the process of training or even extended (with the rule, that all data must be used), the training time has been extended several times.

Our proposition is not the best one for using in Internet of Things, but it has many advantages, and above all, flexible functionality when it comes to learning itself, and there is no requirement to train for one specific error. It has been shown that we can start by getting small errors and adding another transaction with better requirement. The more computational equipment is used, the more rewards it receives. And rewards indicate that it can be used more quickly in the future so it is possible to consume or failure of used device.

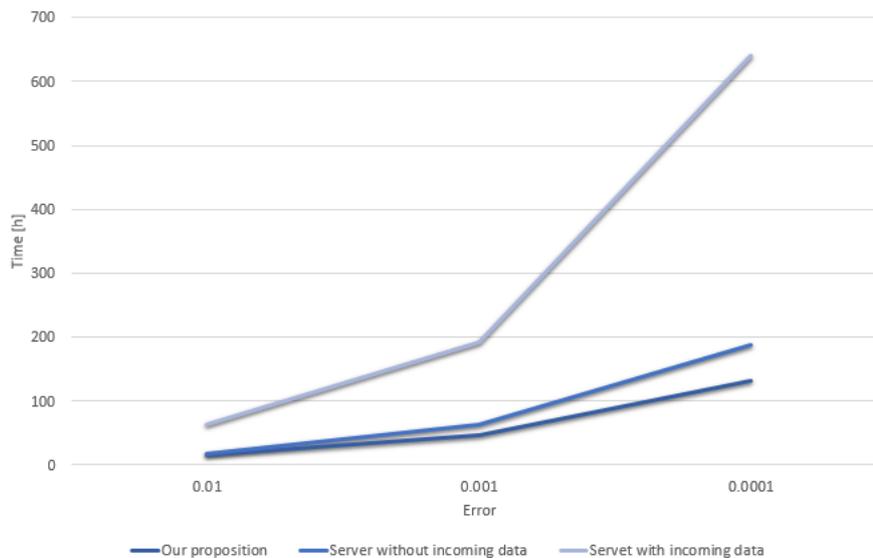


Figure 5. Time needed to train classifier (with 4 hidden layers) to particular error.

6. Conclusions

The introduction of the 5G network will enable the rapid development of the Internet of Things. And above all, it will enable communication in every place with the help of various objects. In order not to lose the privacy of private data, it is worth creating an architecture of operation using artificial intelligence methods in local devices. As local, it is understood all device of one user which can communicate with each other and exchange data. It should guarantee the privacy of all user data.

To make this possible, it would be necessary to perform all operations on given devices, without external servers. That is why in this paper, we presented a solution that uses blockchain idea. We have modified the idea in such a way as to enable the training of classifiers on local devices. Such a solution does not need an Internet connection, but any other one just to be able to communicate a short distance with other devices in a given user network. For the work done, i.e., the training of the classifier, the device is rewarded. The prize in the meaning of the blockchain operation is the appropriate wear indicator for the given equipment. The higher this value, the more likely the equipment will fail. This solution allows to pay attention to possible problems in the future.

Author Contributions: Conceptualization, A.W. and K.K.; methodology, A.W. and K.K.; software, A.W. and K.K.; writing—original draft preparation, A.W. and K.K.; writing—review and editing, A.W. and K.K.

Funding: This research received no external funding.

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

References

1. Samek, W.; Wiegand, T.; Müller, K.R. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv* **2017**, arXiv:1708.08296
2. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **2016**, *529*, 484. [[CrossRef](#)] [[PubMed](#)]
3. He, Y.; Yu, F.R.; Zhao, N.; Leung, V.C.; Yin, H. Software-defined networks with mobile edge computing and caching for smart cities: A big data deep reinforcement learning approach. *IEEE Commun. Mag.* **2017**, *55*, 31–37. [[CrossRef](#)]
4. Kibria, M.G.; Nguyen, K.; Villardi, G.P.; Zhao, O.; Ishizu, K.; Kojima, F. Big Data Analytics, Machine Learning and Artificial Intelligence in Next-Generation Wireless Networks. *IEEE Access* **2018**, *6*, 32328–32338. [[CrossRef](#)]

5. Ahmmed, R.; Swakshar, A.S.; Hossain, M.F.; Rafiq, M.A. Classification of tumors and it stages in brain MRI using support vector machine and artificial neural network. In Proceedings of the International Conference on Electrical, Computer and Communication Engineering (ECCE), Cox's Bazar, Bangladesh, 16–18 February 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 229–234.
6. Todoroki, Y.; Han, X.H.; Iwamoto, Y.; Lin, L.; Hu, H.; Chen, Y.W. Detection of liver tumor candidates from CT images using deep convolutional neural networks. In *International Conference on Innovation in Medicine and Healthcare*; Springer: Cham, Switzerland, 2017; pp. 140–145.
7. Sun, W.; Liu, J.; Yue, Y.; Zhang, H. Double Auction-Based Resource Allocation for Mobile Edge Computing in Industrial Internet of Things. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4692–4701. [[CrossRef](#)]
8. Pace, P.; Aloï, G.; Gravina, R.; Caliciuri, G.; Fortino, G.; Liotta, A. An Edge-based Architecture to Support Efficient Applications for Healthcare Industry 4.0. *IEEE Trans. Ind. Inform.* **2018**, *15*, 481–489. [[CrossRef](#)]
9. Patel, P.; Ali, M.I.; Sheth, A. On using the intelligent edge for IoT analytics. *IEEE Intell. Syst.* **2017**, *32*, 64–69. [[CrossRef](#)]
10. Chen, X.; Shi, Q.; Yang, L.; Xu, J. ThriftyEdge: Resource-Efficient Edge Computing for Intelligent IoT Applications. *IEEE Netw.* **2018**, *32*, 61–65. [[CrossRef](#)]
11. Sharma, P.K.; Chen, M.Y.; Park, J.H. A software defined fog node based distributed blockchain cloud architecture for IoT. *IEEE Access* **2017**, *6*, 115–124. [[CrossRef](#)]
12. Miller, D. Blockchain and the Internet of Things in the Industrial Sector. *IT Prof.* **2018**, *20*, 15–18. [[CrossRef](#)]
13. Governatori, G.; Idelberger, F.; Milosevic, Z.; Riveret, R.; Sartor, G.; Xu, X. On legal contracts, imperative and declarative smart contracts, and blockchain systems. *Artif. Intell. Law* **2018**, *26*, 377–409. [[CrossRef](#)]
14. Zhang, Y.; Wen, J. The IoT electric business model: Using blockchain technology for the internet of things. *Peer Netw. Appl.* **2017**, *10*, 983–994. [[CrossRef](#)]
15. Li, Z.; Kang, J.; Yu, R.; Ye, D.; Deng, Q.; Zhang, Y. Consortium blockchain for secure energy trading in industrial internet of things. *IEEE Trans. Ind. Inform.* **2018**, *14*, 3690–3700. [[CrossRef](#)]
16. Sikorski, J.J.; Haughton, J.; Kraft, M. Blockchain technology in the chemical industry: Machine-to-machine electricity market. *Appl. Energy* **2017**, *195*, 234–246. [[CrossRef](#)]
17. Brousmiche, K.L.; Durand, A.; Heno, T.; Poulain, C.; Dalmieres, A.; Hamida, E.B. Hybrid cryptographic protocol for secure vehicle data sharing over a consortium blockchain. *Proc. IEEE Blockchain* **2018**, *2018*.₂₀₁₈.00223. [[CrossRef](#)]
18. Aitzhan, N.Z.; Svetinovic, D. Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams. *IEEE Trans. Dependable Secur. Comput.* **2018**, *15*, 840–852. [[CrossRef](#)]
19. Dorri, A.; Kanhere, S.S.; Jurdak, R. Towards an optimized blockchain for IoT. In Proceedings of the Second International Conference on Internet-of-Things Design and Implementation, Pittsburgh, PA, USA, 18–21 April 2017; ACM: New York, NY, USA, 2017; pp. 173–178.
20. Kuo, T.T.; Zavaleta Rojas, H.; Ohno-Machado, L. Comparison of blockchain platforms: A systematic review and healthcare examples. *J. Am. Med Inform. Assoc.* **2019**, *26*, 462–478. [[CrossRef](#)] [[PubMed](#)]
21. Macdonald, M.; Liu-Thorold, L.; Julien, R. The blockchain: A comparison of platforms and their uses beyond bitcoin. *COMS4507-Adv. Comput. Netw. Secur.* **2017**. [[CrossRef](#)]
22. McCulloch, W.S.; Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* **1943**, *5*, 115–133. [[CrossRef](#)]
23. Van Ooyen, A.; Nienhuis, B. Improving the convergence of the back-propagation algorithm. *Neural Netw.* **1992**, *5*, 465–471. [[CrossRef](#)]

