Performance Comparison of Multiple ANN Optimizer on IoT-enabled Sensor Fire Dataset

Sudip Suklabaidya
Department of Computer Science,
Assam University Silchar, India
sudipbaidya@gmail.com

Indrani Das
Department of Computer Science,
Assam University Silchar, India
indranidas2000@gmail.com

Abstract: *In today's world, fires in homes and commercial places are a serious problem that can harm the local environment as well as jeopardize people's property and lives. This study predicts the sensor dataset gained from an integrated sensor framework with an artificial neural network. The major goal of this research was to identify a convenient way to encode input data that balanced information loss with simplicity. This paper developed an Artificial Neural Network (ANN) model and applied it to the fire dataset collected from the Integrated Sensor System (ISS). Every neuron of the model will learn and hold weights that weigh information, which provides better accuracy. To mitigate loss functions and improve accuracy, various activation functions such as Sigmoid, Relu, and optimizer Stochastic Gradient Descent (SGD), Adam, and Adamax are used in the designed model. The results demonstrated that the prediction accuracy of the ANN model with Adam as the optimizer is better than that of the other two optimizers. The findings also show that the ANN model performs well in terms of prediction accuracy and is also better suited to the sensor fire dataset.*

Keywords: *Sensor dataset, fire detection, ANN, SGD, adam, adamax.*

Received August 9, 2021; accepted March 6, 2023
https://doi.org/10.34028/iajit/20/5/9

1. Introduction

An Artificial Neural Network (ANN) is a mathematical model that can be easily constructed in a software simulator to replicate two crucial features of the human brain in terms of its high parallel information processing capabilities. The first attribute shows a capacity to learn from examples. The second characteristic is the capacity to generalize the knowledge acquired throughout the learning process to new datasets and anticipated datasets [2]. Fire alarm systems are now an essential part of all construction types due to the rapid increase in the frequency of fire incidents. Fire outbreaks have been listed as one of the disasters that we encounter more frequently when we think about disasters. Numerous techniques for early fire detection have been proposed and put into practice to reduce such accidents. Most of the techniques have thus far produced the desired results. However, because early fire detection is ineffective at preventing the activation of false alarm systems, it does not seem to be a promising solution. Current fire detection technologies rely on physical sensors like heat detectors, smoke detectors, and flame detectors. The reliability of the results produced by these sensor-based detection systems is questionable [7]. The advantages that existed with physical sensor-based systems in the past have been minimized by major highlighted areas in artificial intelligence, such as vision-based study disciplines like Image Processing and Computer Vision, which have produced distinct results. Since neural-based fire detection has outperformed physical sensor-based systems in terms of cost, accuracy, robustness, and reliability, it has become the preferred method over those systems. A new model can be put into practice, taking robustness and reliability measures into consideration, with such advancements in technology and a great vision for neural-based fire detection techniques. We have cited a model that combines the strengths of the existing fire-based detection system and the neural-based fire detection approach, resulting in more accurate results while also reducing the common false triggering in the physical sensor-based system. The contribution of this study is discussed as follows:

We have introduced a model that seems more promising with the blended approach, as mentioned above. This model, which runs on the Arduino Node MCU and uses IoT for real-time fire detection, has the best fire detection performance while maintaining a respectable frame rate.

We used the aforementioned incorporation to create our neural network model, which we assessed and trained using data from the integrated fire sensor system. The model is tested on our machine-generated dataset with different optimizers.

The following is a breakdown of the paper's structure: The second part delves into previous research in sensor-based fire detection, machine learning, and artificial neural networks. Part 3 describes the strategy and in part 4 illustrates the observations of various optimizers along with the findings. Finally, in section 5, there are some final interpretations.

2. Literature Review

ANN and logistic regression were used by Bisquert *et al*. [4] to estimate forest fires using data from remote sensing and fire history. Neural networks have been shown to provide better accuracy and precision in all applied input combination scenarios. Additionally, classification is carried out in the study at different levels after neural network results are obtained, which is helpful and makes deterrence and destruction tasks easier. Also, Safi and Bouroumi [13] use ANN to predict forest fires. They have chosen multilayer perception as their architectural framework. The back-propagation algorithm, which minimizes the overall error at the output layer and converges to a local minimum, is used to train the network. They used the C++ programming language to create the application, which is now being used in the real world to combat forest fires. Hussein *et al*. [5] developed a smart home prototype that allows disabled people to be self-sufficient and secure in their daily activities. They used feed-forward and recurrent neural networks to predict their next move in the house, calculate pre-alarm notifications, and keep alert of their surroundings. Satir *et al*. [15] used multilayer perception based on a back-propagation algorithm to estimate the likelihood of forest fires in Mediterranean forestland. The results are satisfactory using Multilayer Perception (MLP) for their small dataset. Still, they proposed that some other parametric techniques may be better, such as logistic regression, which can be applied to a sufficient dataset. Neves *et al*. [11] describe a machine learning-based model-free damage detection methodology. In the first phase of their method, they applied ANN to the input data gathered from a healthy bridge. The Gaussian process is used in the second step to measure the network's expected mistakes statistically, which aids in the decision-making process for a damage detection threshold. Later, they used the Receiver Operating Characteristic curve (ROC) to determine the true and false positive detections. Manickam *et al*. [10] focuses on the importance of ANN in problem formulation for student performance prediction. They used various ANN techniques, of which Dragonfly algorithms gave better results than other techniques used in the study. They also proposed that the e-learning process should incorporate neural network design to improve academic teaching and learning processes. Awan *et al*. [3] use artificial neural networks to forecast heart disease. Here, the author used a number of techniques in ANN and found accuracy at 94%, but after the feature scaling method, which is with Principal Component Analysis (PCA) help, the accuracy improved to 97%. Al-Janabi *et al*. [1] also work on the prediction of forest fires through various soft computing approaches. In the first stage, they used PCA for critical patterns and the particle swarm optimization methodology for clustering the fire zones. In the second stage, they used multiple soft computing approaches based on neural networks in parallel to determine the best strategies for predicting forest fires accurately and optimally. Later on, they evaluate performance by applying various qualitative metrics. They found that the support vector machine delivers the best predictions compared to others with a lower estimation error. Al-Shawwa *et al*. [2] work on predicting temperature and humidity using ANN. They also used a multilayer perception model and found that the ANN model provided 100% accuracy. In their work on a diagnostic system for diagnosing heart illness, Subhadra *et al*. [17] discovered that multilayer sensory neural networks outperform other classification techniques. The back-propagation algorithm is repeated until they receive the minimum error rate. The dataset of breast cancer patients' data used by Saritas and Yasar [14] is also used. As compared to the Naive Bayes classification algorithm, they found that ANN predicts good results in this study. To predict forest fires in the area. Li *et al*. [9] used Support Vector Machine (SVM) with a radial basis function and a multilayer feed-forward artificial neural network. The back-propagation ANN, they discovered, outperforms the SVM in terms of accuracy and estimation error. ANN prediction models are offered by the authors; Niazkar and Niazkar [12], to help policymakers in various nations predict the COVID-19 epidemic. The results indicate that the implementation of the ANN model gave a more accurate estimation during the incubation period of the COVID-19 outbreak. Also, the model predicts the approximate number of daily cases in the peak countries.

3. Working Methodology

The methodology's primary approach is as follows, as shown in Figure 1.

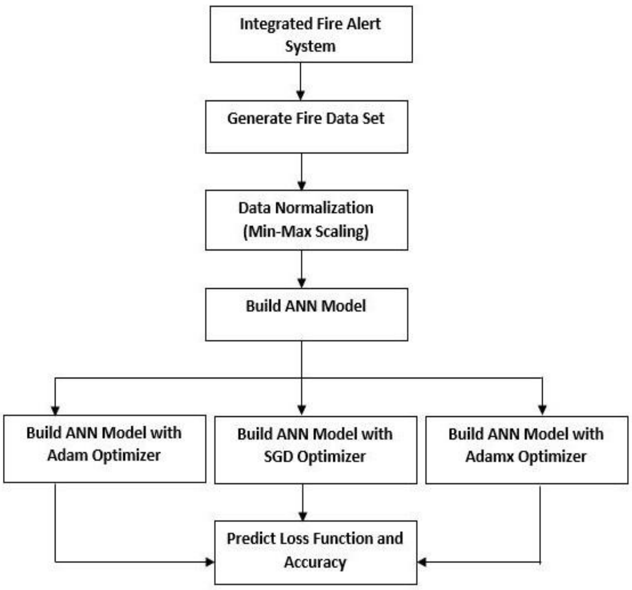


Figure 1. Working mechanism of the methodology.

1. A viable implementation of a comprehensive fire warning system using the Internet of Things (IoT)

and various sensors has been established to substitute traditional sensor-based fire detection alarm systems.
2. A fire dataset was generated from the fire detection unit, which was then normalized using min-max scaling.
3. AN ANN model has been built with different optimizers that use the fire dataset to predict the loss function, validation, and accuracy in the training and testing phases.

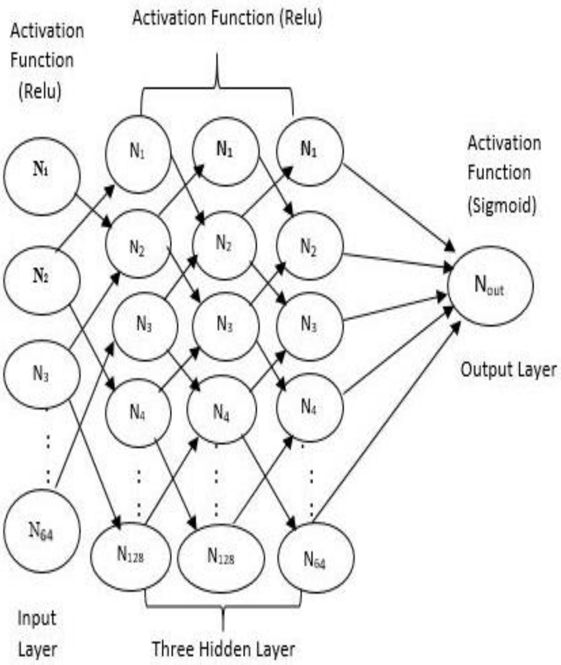


Figure 2. General description of the functioning of the ANN.

A neural network simulates the human brain by learning from a training dataset and then using that knowledge to simplify patterns for classification and prediction. ANN is an information processing paradigm that is based on the biological nervous system and consists of a network of unified neurons. The data from the integrated sensor framework is supplied into the neural network's input layer, as shown in Figure 2. The model uses a perceptron to process data by passing it from the input layers to the hidden layer and then to the output layer. Each input is given a random weight as it passes from the input layer to the concealed layer. The inputs are then multiplied by their weights, and the sum is passed through the network. Every perceptron is given a numerical value known as bias. Additionally, each perceptron is subjected to activation or a transformation function, which decides whether or not a perceptron is activated. Data is transmitted to the next layer using an activated perceptron. Depending on the applications or model's requirements, the type of activation function may alter for different layers. The data is carried forward in this way until the perceptron reaches the output layer of the neural network. At the output layer, a probability is calculated to determine whether the data is classified as Class A (fire) or Class B (nonfire). In such a case, the neural network is trained using back-propagation if the expected output is incorrect. The neural network weights are initially initialized to each input using some input values when designing. The relevance of each input variable is indicated by the weights. Therefore, if we propagate backward in a neural network and compare the actual output to the predicted output, the weights of each input layer can be readjusted so that the loss value can be minimized. This results in a more accurate output, and thus a neural network produces much better results than the traditional ML algorithm.

4. Observation and Illustration

The model developed above is implemented using different optimizers [6] and different activation functions. The ANN model with different optimizers is illustrated below with a confusion matrix and classification report. Further, the loss function and accuracy of each optimizer are represented graphically.

4.1. ANN model with Optimizer SGD (Stochastic Gradient Descent)

Gradient Descent (GD) takes a complete dataset to process. Processing a huge dataset, takes more processing power. Also, to load and update weights, GD needs more resources and is computationally expensive. To prevent this, SGD is used, which takes one record at a time instead of taking the whole record. Every record that takes time will undergo forward and backward propagation, which also takes much more time. The convergence will be much slower to reach the specific point. The formula for the weights, bias, and loss functions is given in Equations (1) and (2). For this mini-batch, SGD came into existence, which will fix some batch sizes. It creates zigzag lines because we are taking a batch of data and don't know the whole data, so the weight update will take a little bit more time compared to GD.

$$\left.\begin{aligned} W_t &= W_{t-1} - \eta \frac{\delta L}{\delta W} \\ B_t &= B_{t-1} - \eta \frac{\delta L}{\delta B} \end{aligned}\right\} \tag{1}$$

$$\text{Loss} = \frac{1}{n} \sum_{i=1}^{n} (Y - \overline{Y})^2 \tag{2}$$

In a dataset of 'n' individual data points or observations, the loss function is the sum of the squared differences between the correct response y, and the calculated response $\overline{Y}$.

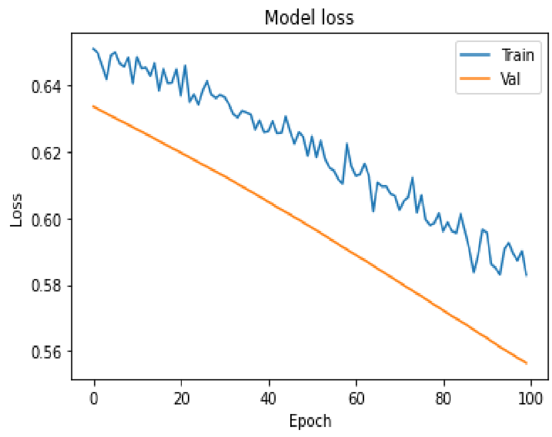
In the mini-batch SGD situation, noise will arise, which needs to be reduced for smooth convergence. To minimize noise, SGD with momentum techniques is applied, where the exponentially weighted average is used with the formulae shown in Equation (3).

$$W_t = W_{t-1} - \eta\, V_{dw} \tag{3}$$

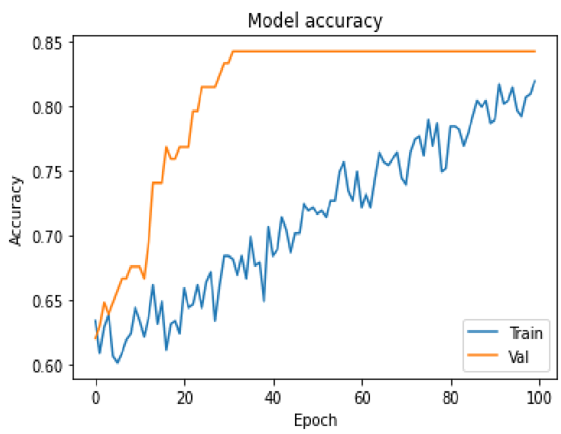
Where, V_{dw} is the exponential weighted average and η is the learning rate.

$$V_{dw_t} = \beta V_{t-1} - \eta \frac{\delta L}{\delta B} \tag{4}$$

Keras is used to create and test the model. The model is trained on a dataset of approximately 300 sample data points with an epoch of 100 and a loss function of binary cross entropy. Figure 3 illustrates the model loss and model accuracy with the SGD optimizer. In addition, the test dataset of around 108 samples is fed into the designed ANN model, which provides true positive and true negative results as shown in the confusion matrix, as well as an accuracy of 86% as shown in the classification report with the SGD optimizer. The SGD optimizer's classification reports, as well as the whole confusion matrix report, are presented in Tables 1 and 2.



a) Loss function with SGD optimizer.



b) Validation accuracy with SGD optimizer.

Figure 3. Efficiency level using SGD optimizer.

Table 1. Confusion matrix of adam optimizer.

	Positive	Negative
Positive	66	11
Negative	4	27

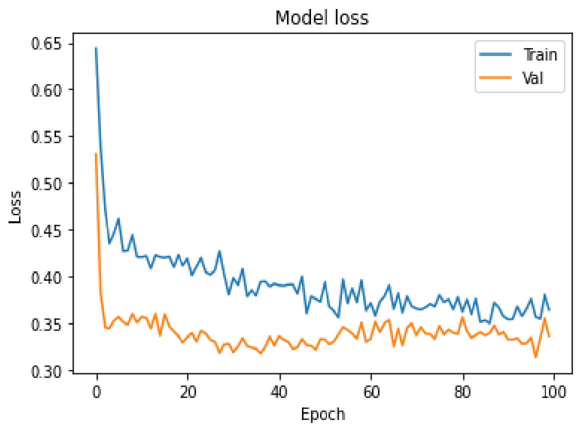
Table 2. Classification report with SGD optimizer.

Label	Precision	Recall	F1 Score	Support	Accuracy
0 (No-Fire)	0.94	0.86	0.90	77	0.86
1 (Fire)	0.71	0.87		31	
Total				108	
Macro Avg.	0.83	0.86	0.84	108	
Weighted Avg.	0.88	0.86	0.86	108	

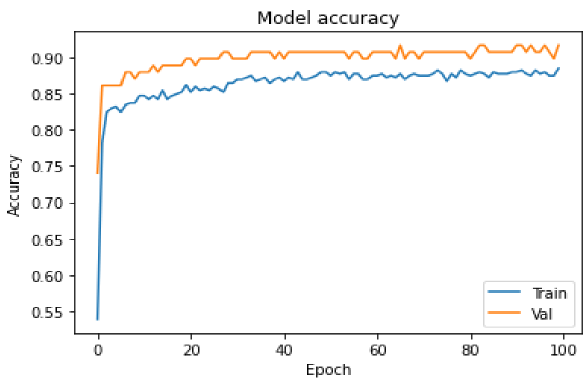
4.2. ANN model with Optimizer Adam

AdaGrad and Root Mean Squared Proportion (RMSProp) have been combined to create Adam [18]. AdaGrad uses the first-order derivative, or normal derivation, and RMSProp uses the square derivative [8, 16]. When we optimize a function, AdaGrad will make the step size bigger, and RMSProp will ensure that when it comes closer to the global minima, it will not drift the weight; that is, it will not over fit the global minima. AdaGrad is an optimizer that adapts parameter-specific learning rates based on how often a parameter is updated during training. The learning rate of a parameter decreases as the number of updates increases. The fundamental downside of AdaGrad is that it continues to decrease during iteration, taking a long time to attain the global minimum. The error function, or cost function, is what the loss function is all about. To reduce the loss function, Adam's optimizer used back-propagation to update all the weights using the formula in Equation (3). To reduce the noise, a major limitation in SGD, the Adam optimizer uses the momentum concept for smoothing the zigzag lines and RMSProp to restrict the fixed learning rate in SGD.

Similarly, the training dataset is sent into the Adam optimizer to determine the loss function and validation accuracy concerning the epoch Tables 3 and 4 shows the confusion matrix and classification report. Figure 4 depicts the loss function, validation, and accuracy of the dataset that is fed into the model.



a) Loss function with epoch.



b) Validation and accuracy with epoch.

Figure 4. Efficiency level using adam optimizer.

Table 3. Confusion matrix with adam optimizer.

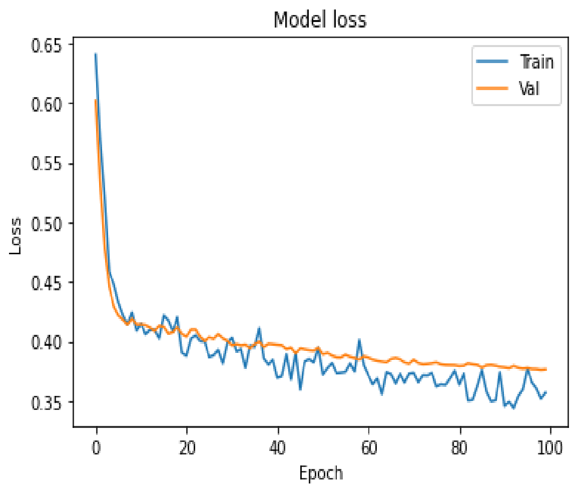
	Positive	Negative
Positive	62	11
Negative	1	34

Table 4. Classification report with adam optimizer.

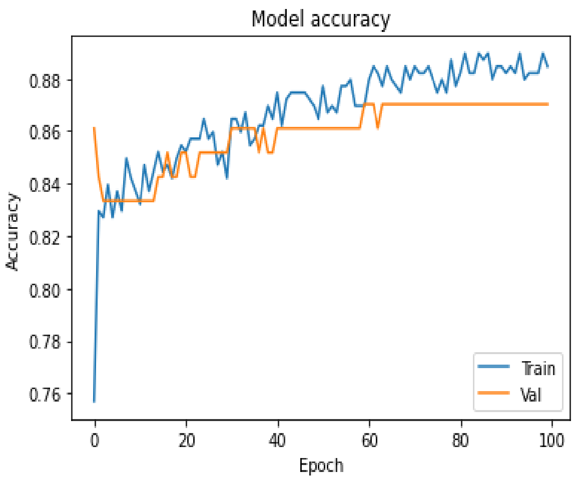
Label	Precision	Recall	F1 score	Support	Accuracy
0 (No-Fire)	0.98	0.85	0.91	77	0.89
1 (Fire)	0.76	0.97	0.85	35	
Total				108	
Macro Avg.	0.87	0.91	0.88	108	
Weighted Avg	0.91	0.89	0.89	108	

4.3. ANN model with Optimizer Adamax

It is a variant of Adam's optimizer. Adamax is based on the L2 infinity norm, which states that the loss value will converge to a single point. It works better for n-dimensional data. The loss function and the Accuracy is depicted in Figure 5. Tables 5 and 6 also include the Adamax optimizer's test data confusion matrix and classification report.



a) Loss function of adamax optimizer with epoch.



b) Accuracy of adamax optimizer with epoch.

Figure 5. Efficiency level using adamax optimizer.

Table 5. Confusion matrix with adamax optimizer.

	Positive	Negative
Positive	51	12
Negative	5	40

Table 6. Classification report with adamax optimizer.

Label	Precision	Recall	F1 score	Support	Accuracy
0 (No-Fire)	0.91	0.81	0.86	63	0.84
1 (Fire)	0.77	0.89	0.82	45	
Total				108	
Macro Avg.	0.84	0.85	0.84	108	
Weighted Avg	0.85	0.84	0.84	108	

5. Conclusions

In general, the universal model, such as machine learning techniques used as a statistical model, is best suited for all cases. But for a specific purpose and for a particular dataset, an ANN model can be used for better accuracy and precision. As fire outbreaks are a vital issue, we move from machine learning techniques to ANN, where it is found that ANN comes out with better accuracy. Here we developed our own ANN model with a different optimizer for the fire dataset that is generated from the sensor framework. It is found that the model works satisfactorily for the particular dataset. The Adam optimizer used in the model gives better accuracy than the other two optimizers used.

References

[1] Al-Janabi S., Al-Shourbaji I., and Salman M., “Assessing the Suitability of Soft Computing Approaches for Forest Fires Prediction,” *Applied Computing and Informatics*, vol. 14, no. 2, pp. 214-224, 2018. https://doi.org/10.1016/j.aci.2017.09.006

[2] Al-Shawwa M., Al-Absi A., Abu Hassanein S., Abu Baraka K., and Abu-Naser S., “Predicting Temperature and Humidity in the Surrounding Environment Using Artificial Neural Network,” *International Journal of Academic Pedagogical Research*, vol. 2, no. 9, pp. 1-6, 2018. http://dstore.alazhar.edu.ps/xmlui/handle/123456789/319

[3] Awan S., Riaz M., and Khan A., “Prediction of Heart Disease Using Artificial Neural Network,” *VFAST Transactions on Software Engineering*, vol. 6, no. 1, pp. 51-61, 2018. DOI: http://dx.doi.org/10.21015/vtse.v13i3.511

[4] Bisquert M., Caselles E., Sánchez J., and Caselles V., “Application of Artificial Neural Networks and Logistic Regression to the Prediction of Forest Fire Danger in Galicia Using MODIS Data,” *International Journal of Wildland Fire*, vol. 21, no. 8, pp. 1025-1029, 2012. DOI:10.1071/WF11105

[5] Hussein A., Adda M., Atieh M., and Fahs W., “Smart Home Design for Disabled People Based on Neural Networks,” *Procedia Computer Science*, vol. 37, pp. 117-126, 2014. https://doi.org/10.1016/j.procs.2014.08.020

[6] Jais I., Ismail A., and Nisa S., “Adam Optimization Algorithm for Wide and Deep Neural Network,” *Knowledge Engineering and*

Data Science, vol. 2, no. 1, pp. 41-46, 2019. DOI:10.17977/um018v2i12019p41-46

[7] Khondaker A., Khandaker A., and Uddin J., "Computer Vision-based Early Fire Detection Using Enhanced Chromatic Segmentation and Optical Flow Analysis Technique," *The International Arab Journal of Information Technology*, vol. 17, no. 6, pp. 947-953, 2020. https://doi.org/10.34028/iajit/17/6/13

[8] Kingma D. and Ba J., "Adam: A Method for Stochastic Optimization," *in Proceedings of the 3rd International Conference for Learning Representations*, San Diego, 2015. https://doi.org/10.48550/arXiv.1412.6980

[9] Li Y., Feng Z., Chen S., Zhao Z., and Wang F., "Application of the Artificial Neural Network and Support Vector Machines in Forest Fire Prediction in the Guangxi Autonomous Region, China," *Hindawi Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1-14, 2020. https://doi.org/10.1155/2020/5612650

[10] Manickam M., Mohanapriya M., Kale S., Uday M., Kulkarni P., Khandagale Y., and Patil S., "Research Study on Applications of Artificial Neural Networks and E-Learning Personalization," *International Journal of Civil Engineering and Technology*, vol. 8, no. 8, pp. 1422-1432, 2017. http://http://www.iaeme.com/ijciet/issues.asp?JType=IJCIET&VType=8&IType=8

[11] Neves A., Gonzalez I., Leander J., and Karoumi R., "Structural Health Monitoring of Bridges: A Model-Free ANN-based Approach to Damage Detection," *Journal of Civil Structural Health Monitoring*, vol. 7, pp. 689-702, 2017. https://doi.org/10.1007/s13349-017-0252-5

[12] Niazkar H. and Niazkar M., "Application of Artificial Neural Networks to Predict the COVID-19 Outbreak," *Global Health Research and Policy*, vol. 5, no. 1, pp. 1-11, 2020. https://doi.org/10.1186/s41256-020-00175-y

[13] Safi Y. and Bouroumi A., "Prediction of Forest Fires Using Artificial Neural Networks," *Applied Mathematical Sciences*, vol. 7, no. 5, pp. 271-286, 2013. DOI:10.12988/ams.2013.13025

[14] Saritas M. and Yasar A., "Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 7, pp. 88-91, 2019. DOI:10.18201//IJISAE.2019252786

[15] Satir O., Berberoglu S., and Donmez C., "Mapping Regional Forest Fire Probability Using Artificial Neural Network Model in a Mediterranean Forest Ecosystem," *Geomatics, Natural Hazards and Risk*, vol. 7, no. 5, pp. 1645-1658, 2016. http://dx.doi.org/10.1080/19475705.2015.108451

[16] Soydaner D., "A Comparison of Optimization Algorithms for Deep Learning," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 34, no. 13, pp. 1-26, 2020. DOI:10.1142/S0218001420520138

[17] Subhadra K. and Vikas B., "Neural Network Based Intelligent System for Predicting Heart Disease," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 5, pp. 484-487, 2019. https://www.ijitee.org/portfolio-item/d2770028419/

[18] Yi D., Ahn J., and Ji S., "An Effective Optimization Method for Machine Learning Based on ADAM," *Applied Sciences*, vol. 10, no. 3, pp. 1073, 2020. DOI:10.3390/app10031073

Sudip Suklabaidya he did his MSc.(Computer Science) and MBA(IT) from Assam University Silchar and currently pursuing PhD in Assam University Silchar, India. His research interests primarily comprises on IoT, Machine learning, Artificial Neural Network, CNN, and optimization methods.

Indrani Das she did her B.E, M.Tech and Ph.D in Computer Science and Engineering. She is working as an Assistant Professor in the Department of Computer Science Assam University (Central University), Silchar, Assam, India. She received her Ph.D from School of Computer and System Sciences, Jawaharlal Nehru University, New Delhi, India in 2015. Her current research interest includes Mobile Adhoc Networks and Vehicular Adhoc Networks, IoT.