Neelamadhab Khaya[1]
Binod Kumar Pattanayak[2]
Bichitrananda Patra[3]
Ahmad Khader Habboush[4]
Bibhuprasad Mohanty[5]

Energy Efficient Predictive Maintenance in Resource Constraint Industrial IoT devices using Smoothed Mish Activation Function

Journal of Electrical Systems

Abstract: - Traditionally, Industrial IoT devices collect sensor data to a cloud platform where ML/DL processing is to be done. Edge computing has the advantage of reducing latency, improved battery performance, safe transmission and reducing vulnerability. These benefits are particularly significant considering the limited resources on IoT devices, such as only a few kilobytes of RAM, and the critical importance of energy savings in industrial applications. Arduino Uno uses 2kB of RAM and 32 kB of read only flash memory. Optimal performance is required for ML inference on EDGE devices to get good accuracies and low latency. This research paper is based on memory optimization in resource constrained devices, focusing on a novel approach that combines the Bonsai tree architecture with a smoothed Mish activation function. While the Bonsai tree architecture has been explored in prior works, this research contributes to the refinement of the activation function, a crucial component in achieving superior test accuracy within limited memory constraints. Although many algorithms have been developed with good accuracy but can't be used in optimal power consumption with low latency and resource constraint like fewer RAM and ROM. A smooth Mish activation function in Bonsai tree algorithm is proposed to enhance accuracy with lowest ML model size, where Bonsai tree is a single,shallow,sparse tree where data is projected in low dimension space. Lowest ML model size consumes least power. We observed the smoothed Mish activation function in the Bonsai tree algorithm outperform with low latency and power consumption.

This paper aims to elucidate the methodology, experimental results, and implications of the modified activation function, shedding light on its potential to revolutionize memory-efficient edge computing in the Industrial IoT network.

Keywords: Edge computing, energy efficiency, IIOT, activation function.

1. INTRODUCTION

The Internet of Things is a network of devices. It has the ability to collect data from many sources like sensors and store it in local server or cloud databases through gateways[1]. Machine learning is an ability of a computer program that can be trained on past data to predict unseen or new data. Deployment of ML models on cloud platforms will give latency due to large IoT devices and their data transmission and processing delays. Thus Edge computing comes into the feature to avoid unnecessary delays and susceptible to attack[2].

Numerous applications have been created to cater to various sectors such as consumer, enterprise, and societal Industrial IoT. These applications span a wide range of uses, encompassing predictive maintenance, intelligent healthcare, and the development of smart cities and housing, among others. Notably, these applications have traditionally adhered to a specific paradigm, primarily due to resource limitations of Industrial IoT devices.

Recent development in ML architecture produced few kB of memory which can evade expensive communication between IoT and Cloud[3]. Edge ML enables energy efficiency in real time analytics. Energy saving is a major concern in Industrial IoT due to the large number of IoT devices. Cloud computing is vulnerable to attack in IoT

[1] [1]Department of Computer Science and Engineering, Institute of Technical Education and Research, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, India, Email: neela.khaya@gmail.com

[2]Department of Computer Science and Engineering, Institute of Technical Education and Research, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, India, Email: binodpattanayak@soa.ac.in

[3]Department of Computer Application, Institute of Technical Education and Research, Siksha 'O' Anusandhan [4]Deemed to be University, Bhubaneswar, India, Email: bichitranandapatra@soa.ac.in

[4]Faculty of Information Technology, Jerash University, Jerash, Jordan., Email: ahmad_ram2001@jpu.edu.jo

[5]Department of Electronics and Communication Engineering, Institute of Technical Education and Research, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, India, Email: bibhumohanty@soa.ac.in

networks[4]. Thus Edge computing is more preferable in Industrial IoT than cloud computing for low latency and vulnerability.

2. RELATED WORK

Many researchers have developed algorithms to reduce the size of ML models to enhance efficiency and deploy on a small kB sized micro controller. They used optimal loss algorithms, different activation and loss functions etc. The paper introduces the Bonsai algorithm. It is a tree-based algorithm developed for efficient prediction on resource-constrained IoT devices like Arduino Uno with limited hardware specifications

Bonsai achieves high prediction accuracy by developing sparse trees with powerful nodes to project data into a low-dimensional space and jointly learning tree and projection parameters. Experimental results demonstrate Bonsai's ability to make rapid predictions on slow microcontrollers, fit in KB of memory, and outperform existing algorithms with lower battery consumption. The algorithm also exhibits generalization to resource constrained settings beyond IoT[5].

The activation function of the neural network has a great contribution to training performance. Some well known activation functions show vanishing or exploding gradient descent during back propagation[6].

Recent studies comparing rectified activation functions (ReLU, Leaky ReLU, PReLU, RReLU) in CNNs for image classification challenge the sparsity assumption, showing non-zero negative slopes consistently improve performance[7]. Deterministic negative slopes (Leaky ReLU, PReLU) tend to overfit small datasets, while RReLU's randomized approach offers better generalization without ensemble methods, achieving 75.68% accuracy on CIFAR-100. These findings underscore the importance of activation function selection and suggest randomized approaches like RReLU as promising alternatives for improving CNN performance.

The research work came up with a new activation function for light weight computer vision tasks. They proposed the Tanh exponential activation function to improve classification tasks[8]. It converges faster and accelerates the learning process as it has a minimum negative value of -0.3532. This activation function is given as

$$f(x) = x.\tanh(e^{x}) \qquad \textbf{(1)}$$

The Gaussian Error Linear Unit (GELU) is a non-convex, non-monotonic activation function that weights inputs based on their values, offering a probabilistic interpretation and increased curvature compared to ReLUs and ELUs[9]. Empirical evaluations demonstrate superior performance of GELUs across various artificial intelligence tasks. GELU activation function defined by

$$f(x) = \frac{x}{2} * (1 + \mathrm{erf}(\frac{x}{\sqrt{2}})) \qquad \textbf{(2)}$$

Self gated activation function (SWISH) shows better performance than ReLU activation function[10].

Swish activation function is defined as follows

$$f(x) = \frac{x}{1+e^{-x}} \qquad \textbf{(3)}$$

This activation function is unbounded above like ReLU, which helps prevent saturation during training. Swish is one-sided bounded at zero, providing stability and regularization effects. The function is smooth, aiding in optimization and generalization. Swish is non-monotonic, allowing it to capture complex patterns effectively.

The paper introduces the Mish activation function. It is a smooth and nonmonotonic activation function. Its formulation is expressed as follows[11]:

$$f(x) = x.\tanh(\ln(1 + e^{x})) \qquad \textbf{(2)}$$

Mish activation function is boundless above and bounded below. It is smooth and non-monotonic. It improves the result in the Neural Network.

3. OBJECTIVE

Objective of the research is to minimize the model size without reducing the accuracy of edge Industrial IoT devices as energy optimization is a major focus in the industrial sector. Better accuracy can be achieved by utilizing different methods. Activation function application in neurons is one way to achieve better accuracy with reducing overall loss.

4. METHODOLOGY

In this study of research we used Bonsai architecture. Training and optimization steps are as followed below.

Tree Model Design: Bonsai utilizes a single shallow sparse tree with powerful nodes to ensure small model size while maintaining prediction accuracy.

Sparse Data Projection: Data is projected into a low-dimensional space for the tree learning, optimizing both model size and accuracy.

Joint Learning of Tree and Projection Parameters: The algorithm jointly learns all tree and projection parameters, rather than doing it in a stepwise or hierarchical manner.

Efficient Implementation: Bonsai algorithm is used to ensure it is resource-efficient and suitable for devices with limited processing power and memory.

Optimization Techniques: Various optimization techniques are employed, including gradient descent with iterative hard thresholding, to learn the model parameters effectively.

Empirical Evaluation: The methodology includes extensive experiments and comparisons with the methods to validate the efficiency and effectiveness of the Bonsai algorithm in resource-constrained environments.

Activation function: An activation function in artificial neural network systems is a mathematical function that determines the output of a neuron or node based on the weighted sum of its inputs[12]. It introduces non-linearity into the network, allowing neural networks to approximate complex functions and learn from data.

Characteristics of activation function:

i. Should be non-linear

ii. Should be differentiable

iii. Computationally inexpensive

iv. Zero centered

v. Non-saturating

Table-1 Different activation functions

Sl.No	Activation functions	Equations	Advantages	Disadvantages
1	Sigmoid	$f(x) = \frac{1}{1 + e^{-x}}$	*Derivable at every point *Bound between 0 to 1 *Clear predictions at points	*Vanishing gradient *Outputs aren't zero centered *Saturates and kills gradients *Computationally expensive
2.	Tanh	$f(x) = \frac{2}{1 + e^{-2x}} - 1$	*Zero centered	*Vanishing gradient *Computationally expensive *Saturates gradients
3.	ReLU	$f(x) = \max(0, x)$	*Computationally effective *Adds non-linearity *No vanishing gradient	*Dying ReLU problem

4.	Leaky ReLU	$f(x) = \max(\alpha x, x)$	*Remove the dying ReLU	
5.	ELU	$f(x) = x \quad x \geq 0$ $\alpha(e^x - 1) \quad x < 0$	*Derivable at 0 *Produces negative outputs *pushes the mean value towards zero	*Computationally expensive
6.	Mish	$f(x) = x.\tanh(\varsigma(x))$ $\varsigma(x) = \ln(1 + e^x)$	*Non-monotonic	*computationally very expensive
7.	Swish	$f(x) = x.\sigma(x)$	*Non-monotonic *Self-gated	*Computationally expensive
8	Softmax	$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}$	*Handle multiple classes *Compute probability	*For output layer only
9.	Smooth Mish(Proposed)	$\mathbf{f(x)} = \frac{\mathbf{x * \tanh(\log(1 + e^{(\alpha * x)}))}}{\mathbf{1 + e^{-\beta * x}}}$	*Smoother than Mish *Solves vanishing gradient problem	*Computationally expensive

Smoothed Mish activation function:

$$\mathbf{f(x)} = \frac{\mathbf{x * \tanh(\log(1+e^{(\alpha * x)}))}}{\mathbf{1+e^{-\beta * x}}} \qquad \textbf{(3)}$$

In this equation, α and β are hyperparameters that you can adjust to control the behavior of the activation function. The function takes an input x and applies the specified operations to calculate the output f(x) Fig(1a).

Derivative of Smoothed Mish activation function:

$$f'(x) = \sigma(x) * (\tanh(\log(1 + e^{\alpha x})) +$$

$$x * a * \mathrm{sech}^2(\log(1 + e^{\alpha x})) * \frac{e^{\alpha x}}{1 + e^{\alpha x}}) *$$

$$(\mathbf{1 - \beta * \sigma(x)}) \qquad \textbf{(4)}$$

Where σ(x) is sigmoid function:$\sigma(x) = \frac{1}{1+e^{-x}}$

sech(x) is hyperbolic secant function: $\mathrm{sech}(x) = \frac{1}{\cosh(x)}$

α, β are the parameters controlling the shape of the Smooth Mish function.

This formula represents the rate of change (slope) of the Smooth Mish activation function with respect to its input x. The inclusion of the sigmoid and hyperbolic secant functions contributes to the smoothness and improved gradient flow properties of the Smooth Mish activation.

Characteristics of Smoothed Mish activation function

We can modify the Mish activation function to improve its derivative characteristics. "Smooth Mish" activation function, which aims to address the vanishing gradient problem by making the derivative smoother. The idea is to introduce sigmoid functions to make the derivative smoother.

5. DATASET In this research study,CWRU bearing fault diagnosis is used to improve accuracy by using Bonsai tree algorithm with smoothed thanh activation function. CWRU vibration dataset has four no of classes. This dataset contains different bearing defects like inner raceway, outer raceway, ball defect, normal. This data was recorded on motor speed of 1797 to 1720 rpm using accelerometer at fan end and drive end. The data was sampled at 12kHz. In the dataset each file contains 120k to 240k sample points.

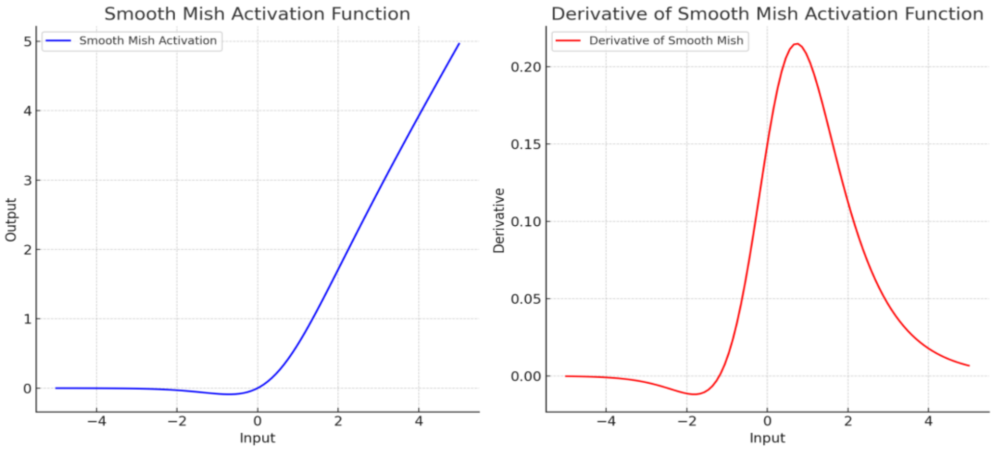

Fig.1a. Smooth Mish activation function,

Fig.1b. Derivative of Mish activation function

Table-2 CWRU bearing fault types

Bearing Type	Class Label	Classes
Outer Ring Damage	OR	Class 0
Inner Ring Damage	IR	Class 1
Ball Damage	B	Class 2

6. ALGORITHMS

This paper uses an optimal smoothed Mish activation function to perform well in accuracy and projected a small size trained model for an energy efficient IIoT system. Base model for training is Bonsai tree architecture. Bonsai tree is a single,shallow,sparse tree where data is projected in low dimension space. It can predict in milliseconds.

Hyper-parameters: Training of each dataset is divided into 80% for training and 20% for validation. The hyperparameters of all algorithms are fixed in the validation set. Once the hyperparameters are determined, the algorithm is trained on the successful training set.

7. RESULTS AND ANALYSIS

7. 1 SIMULATION PROCEDURE

CWRU bearing fault diagnosis dataset has been applied to the Bonsai tree algorithm with Smoothed Mish activation function. Performance metrics were calculated using the python Sklearn package. Confusion matrix plotted with three class labels outer ring(Class–0),inner ring(Class-1),ball (Class-2) damage in Fig 1.
AUC-ROC curve of three classes was plotted as false positive rate vs true positive rate in Fig 2.

It can be seen that the proposed activation function shows best accuracy among latest adopted activation functions(Table-2).

Table-2 Performance comparison table of different activation functions

Activation functions	Test Accuracy in%	Model size in KB	Total loss
ReLU	77.97	25.01	0.7678
LeakyReLU	78.24	25.01	0.7538
Swish	78.25	25.01	0.5700

Mish	77.97	25.01	0.6421
Smoothed Mish(Proposed)	**78.93**	**25.01**	**0.5139**

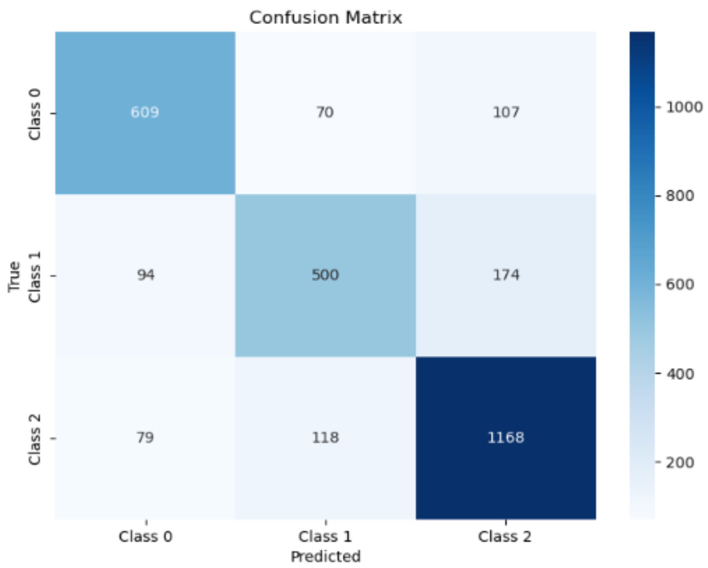

Fig.2. Confusion matrix of CWRU dataset of three classes (Inner race,Outer race,Ball fault)

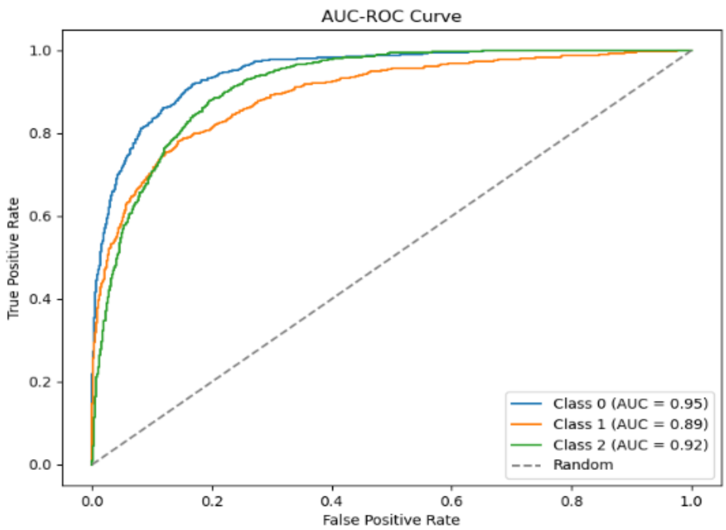

Fig.3. AUC-ROC curve of three classes (Inner race,Outer race,Ball fault)

8. CONCLUSIONS

This research has substantiated Bonsai's exceptional performance across key dimensions of accuracy, efficiency, and model size, demonstrating its ability to thrive within the stringent memory limitations inherent to devices such as the Arduino Uno. The proposed smoothed Mish activation function represents a significant contribution to this field, as it optimizes the trade-off between model accuracy and memory efficiency. Traditional activation functions, such as the widely-used ReLU (Rectified Linear Unit) and sigmoid functions, can be computationally expensive and memory-intensive, making them less suitable for devices with limited resources. In contrast, the smoothed Mish activation function strikes a delicate balance between computational simplicity and model expressiveness. It not only minimizes the computational overhead but also ensures that the model's accuracy remains high, even when operating within the tight memory constraints of edge computing devices in IIoT.

Acknowledgements.

REFERENCES

[1] Bibhuti Bhusana Behera, Rajani Kanta Mohanty, Binod Kumar Pattanayak* 2022. A synthesized architecture and future research directions for Industrial IoT in the mining industry. Journal of East China University of Science and Technology. 65, 2 (Jun. 2022), 511–528.

[2] Behera, B. B., Pattanayak, B. K., & Mohanty, R. K. (2022). Deep Ensemble Model for Detecting Attacks in Industrial IoT. International Journal of Information Security and Privacy (IJISP), 16(1), 1-29. http://doi.org/10.4018/IJISP.311467.

[3] Rath, M. and Pattanayak, B. (2019), "Technological improvement in modern health care applications using Internet of Things (IoT) and proposal of novel health care approach", International Journal of Human Rights in Healthcare, Vol. 12 No. 2, pp.148-162.https://doi.org/10.1108/IJHRH-01-2018-0007.

[4] Hosenkhan, M.R., Pattanayak, B.K. (2020). Security Issues in Internet of Things (IoT): A Comprehensive Review. In: Patnaik, S., Ip, A., Tavana, M., Jain, V. (eds) New Paradigm in Decision Science and Management. Advances in Intelligent Systems and Computing, vol 1005. Springer, Singapore.https://doi.org/10.1007/978-981-13-9330-3_36.

[5] Ashish Kumar, Saurabh Goyal, and Manik Varma. 2017. Resource-efficient machine learning in 2 KB RAM for the internet of things. In Proceedings of the 34th International Conference on Machine Learning - Volume 70 (ICML'17). JMLR.org, 1935–1944.

[6] Hayou, Soufiane, Arnaud Doucet, and Judith Rousseau. "On the selection of initialization and activation function for deep neural networks." arXiv preprint arXiv:1805.08266 (2018).

[7] Xu, Bing, Naiyan Wang, Tianqi Chen, and Mu Li. "Empirical evaluation of rectified activations in convolutional network." arXiv preprint arXiv:1505.00853 (2015).

[8] Liu, Xinyu & di, Xiaoguang. (2020). TanhExp: A Smooth Activation Function with High Convergence Speed for Lightweight Neural Networks.

[9] Hendrycks, D. and Gimpel, K., 2016. Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415.

[10] Ramachandran, Prajit, Barret Zoph, and Quoc V. Le. "Swish: a self-gated activation function." arXiv preprint arXiv:1710.05941 7 (2017).

[11] Misra, D. (2019). Mish: A self regularized non-monotonic activation function. arXiv preprint arXiv:1908.08681.

[12] Liu, Qian, and Steve Furber. "Noisy Softplus: a biology inspired activation function." In International Conference on Neural Information Processing, pp. 405-412. Springer, Cham, 2016.

[13] Exploring the Internet of Things (IoT) in Education: A Review,

[14] Behera, Bibhuti Bhusana; Mohanty, Rajani Kanta; Pattanayak, Binod Kumar. NeuroQuantology; Bornova Izmir Vol. 20, Iss. 6, (2022): 1399 - 1409. DOI:10.14704/nq.2022.20.6.NQ22135.