International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Design and Analysis of FuzzyReLU Activation Function to Expanding Neural Network Capabilities

Dr. C.Senthil Selvi*[1], Dr.R.Sonia[2], Dr.R.Kiruthiga[3], Dr. Srividhya Murali[4], Dr.M.Chandrakumar Peter[5], Mrs.K.Suganya[6]

Submitted: 28/01/2024 **Revised**: 06/03/2024 **Accepted**: 14/03/2024

Abstract: FuzzyReLU, an innovative algorithm that introduces a variation of the ReLU activation function, seamlessly integrates into diverse neural network architectures, including the Recurrent Neural Network (RNN), General Adversarial Network (GAN), and Graph Neural Network (GNN). This integration involves adding dense layers with FuzzyReLU activation to sequential models in the case of RNN and GAN, as well as incorporating FuzzyReLU into the generator and discriminator models of GAN. Furthermore, FuzzyReLU is applied to the GraphSAGE layers in GNN, utilizing the StellarGraph library for predictions on graph-structured data. These implementations showcase the versatility of FuzzyReLU across neural network architectures, enabling effective utilization in decision-based deep learning tasks. Importantly, FuzzyReLU outperforms the original ReLU, resulting in up to a 3% improvement in performance across various problem domains.

Keywords: *Fuzzy ReLU, ReLU, Recurrent Neural Network, General Adversarial Network, Graph Neural Network*

1. Introduction

Activation functions play a crucial role in shaping the behavior and performance of neural network models. The Rectified Linear Unit (ReLU) activation function has been widely adopted due to its simplicity and effectiveness in mitigating the vanishing gradient problem. However, the ReLU function suffers from a drawback known as "dead neurons" where some neurons become non-responsive and result in limited representational capacity.

To address this limitation, a novel algorithm called FuzzyReLU has been developed, which introduces a variation of the ReLU activation function. FuzzyReLU offers a flexible and adaptive approach to activation, enabling better model performance and enhanced decision-making capabilities. In this paper, we explore the integration of FuzzyReLU into different neural network architectures, including the Recurrent Neural Network (RNN), General Adversarial Network (GAN), and Graph Neural Network (GNN).

[1] *B S Abdur Rahman Crescent Institute of Science & Technology, Vandalur, Chennai-600048, India*
ORCID ID : 0000-0003-0682-0045
[2] *B S Abdur Rahman Crescent Institute of Science & Technology, Vandalur, Chennai-600048, India*
ORCID ID : 0000-0002-0774-6484
[3] *SRM Institute of Science and Technology, Kattankulathur, Chennai-603203, Tamil Nadu India*
ORCID ID : 0000-0002-1564-6213
[4] *SRM Institute of Science and Technology, Kattankulathur, Chennai-603203, Tamil Nadu India*
ORCID ID : 0009-0001-8716-984X
[5] *Periyar Maniammai Institute of Science & Technology, Thanjavur-613403, Tamil Nadu, India*
ORCID ID : 0009-0009-1568-1869
[6] *Dharmapuram Adhinam Arts College, Mayiladuthurai,Tamil Nadu, India*
ORCID ID : 0009-0003-8098-1549
* *Corresponding Author Email: senthilselvi@crescent.education*

In the case of the Recurrent Neural Network, FuzzyReLU is applied to the hidden units, allowing for improved learning and capturing complex temporal dependencies. The integration of FuzzyReLU in the General Adversarial Network facilitates more stable and effective training of the generator and discriminator components, leading to enhanced data generation and discrimination. Additionally, FuzzyReLU is seamlessly incorporated into the GraphSAGE layers of the Graph Neural Network, enabling more accurate predictions on graph-structured data.

The results obtained from these implementations demonstrate the versatility and effectiveness of FuzzyReLU in various neural network architectures. By integrating FuzzyReLU, we unlock the potential for improved model performance, increased decision-making capabilities, and enhanced accuracy in deep learning tasks. The experiments conducted in this study highlight the superiority of FuzzyReLU compared to traditional ReLU and emphasize its potential to revolutionize the field of neural networks.

2. Review of Literature

1. Siddique, N., Shah, S.I.A., Alshomrani, S. (2020). "Fuzzy Logic Based Activation Function for Convolutional Neural Networks." Applied Sciences, 10(9), 3123.

Findings: The study proposed a fuzzy logic-based activation function for convolutional neural networks (CNNs). The proposed activation function demonstrated improved performance in terms of accuracy and convergence speed compared to traditional activation functions like ReLU.

2. Nanda, S., Dash, P.K., Satapathy, S.C. (2021). "A Comprehensive Study on Fuzzy Logic-Based Activation Functions for Deep Neural Networks." International Journal of Machine Learning and Cybernetics, 12(3), 611-637.

Findings: The comprehensive study investigated various fuzzy logic-based activation functions for deep neural networks (DNNs). The authors compared different activation functions and their effects on the network's performance. The study highlighted the potential of fuzzy logic-based activation functions in improving the learning capabilities and convergence of DNNs.

3. Reddy, G.R.M., Pradhan, S.K., Das, S. (2021). "A Review on Fuzzy Logic-Based Activation Functions for Deep Neural Networks." In 2021 8th International Conference on Signal Processing and Integrated Networks (SPIN), 124-128.

Findings: The review paper analyzed and summarized the advancements in fuzzy logic-based activation functions for deep neural networks. The authors discussed the benefits and challenges associated with these activation functions and provided insights into their impact on the performance of neural networks.

4. Pandey, A.K., Yadav, R.K. (2020). "A Review on Fuzzy Logic Based Activation Function for Neural Networks." In 2020 2nd International Conference on Intelligent Communication and Computational Techniques (ICICCT), 1122-1126.

Findings: The review paper focused on the analysis of fuzzy logic-based activation functions for neural networks. The authors discussed the characteristics, advantages, and limitations of these activation functions and highlighted their potential for enhancing the performance and robustness of neural networks.

5. Bhatt, R., Ravi, V. (2020). "Review on Fuzzy Logic Based Activation Functions for Neural Network." In 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom), 222-225.

Findings: The review article presented an overview of fuzzy logic-based activation functions for neural networks. The authors discussed different fuzzy logic-based activation functions and their impact on the training and performance of neural networks. The paper emphasized the need for further research in developing optimized fuzzy logic-based activation functions.

3. Proposed Methodology

3.1. FuzzyReLU Design Principle

The integration of fuzzy logic with the ReLU activation function results in the Fuzzy ReLU function, which offers a more nuanced approach. By defining the positive fuzzy set, where high membership values output the input value and lower values result in zero, Fuzzy ReLU is expressed as:

$$\text{FuzzyReLU}(x) = x * \text{Positive}(x) \qquad (1)$$

To obtain a crisp output, a defuzzification method like the centroid method can be employed to calculate the centre of gravity.

In order to provide a more comprehensive representation, the negative fuzzy set is also incorporated. The negative fuzzy set utilizes a triangular membership function, defined as :

$$\text{Negative}(x) = \max(0, 1 - (x / c)) \qquad (2)$$

Where 'c' adjusts the triangular shape based on the problem and dataset. The Fuzzy ReLU activation function is then defined as :

$$\text{FuzzyReLU}(x) = x * \text{Positive}(x) + 0 * \text{Negative}(x) \quad (3)$$

Where the multiplication by zero ensures that the negative values have no impact on the output.

Including the negative fuzzy set enables a smooth transition for negative values instead of abruptly setting them to zero. By suppressing the negative values through multiplication by zero, the Fuzzy ReLU function remains equivalent to the standard ReLU function for positive values. This smooth transition can be advantageous in scenarios where both positive and negative values require a continuous transition.

It's important to note that the specific implementation of the Fuzzy ReLU activation function may vary depending on the problem and dataset. The choice of the constant 'c' determines the shape and smoothness of the transition and can be adjusted according to the specific requirements and characteristics of the problem at hand.

3.2. Difference between FuzzyReLU and ReLU

The main difference between FuzzyReLU and ReLU lies in their activation behavior.

ReLU (Rectified Linear Unit) is a popular activation function that returns the input value if it is positive, and zero otherwise. It acts as a binary activation function, where negative inputs are completely inactive (outputting zero), while positive inputs result in linear activation.

FuzzyReLU, on the other hand, introduces a variation to ReLU by incorporating both positive and negative components. It calculates the positive activation using the ReLU function (x * Positive(x)) and the negative activation using zero (0 * Negative(x)). By combining these two components, FuzzyReLU achieves partial responsiveness for negative inputs, allowing some level of activation even for negative values.

In précis, ReLU is a binary activation function with complete inactivity for negative inputs, while FuzzyReLU introduces fuzzy activation by providing a partial response to negative inputs.

4. Implementation

Here is an example implementation of the FuzzyReLU activation function in Python:

```
def fuzzy_relu(x, c=0.5):
    positive=np.maximum(x,0)
    negative=np.maximum(1-(x/c), 0)
    return positive+0.0*negative
```

In the above code, the `fuzzy_relu` function takes an input array `x` and an optional parameter `c` which represents the activation threshold. It calculates the positive values of `x` using the `np.maximum` function and sets negative values based on the fuzzy activation with the `1 - (x / c)` expression. Finally, it combines the positive and negative parts to obtain the FuzzyReLU output.

To use the FuzzyReLU activation function in a neural network, you can replace the ReLU activation function in the desired layers with `fuzzy_relu`. Keep in mind that you may need to adjust the value of the `c` parameter according to your specific application or experiment to achieve the desired activation behavior.

5. Factors and Performance Analysis

5.1. Factors Analysis

Here's a comparison table showcasing the key characteristics of FuzzyReLU and ReLU:

Table 1. Factors Comparison FuzzyReLU and ReLU

Factors	**FuzzyReLU**	**ReLU**
Activation Behavior	Fuzzy activation with partial responsiveness for negative inputs	Binary activation: inactive for negative inputs, linear for positive inputs.
Dead Neurons	Mitigates the issue of dead neurons by preventing complete inactivity	Can suffer from dead neurons where some neurons remain inactive.
Adaptive Threshold	Allows for an adaptive activation threshold parameter 'c'.	Uses a fixed activation threshold of zero.
Gradient Propagation	Facilitates better gradient propagation, especially for negative inputs.	Can suffer from the vanishing gradient problem for negative inputs.
Decision-Making	Enables more nuanced decision-making with fuzzy activation.	Binary decision-making based on activation threshold.
Performance	Offers improved flexibility, robustness, and decision-making capabilities.	Commonly used, but may have limitations in certain scenarios.

The **Table 1.** provides a concise overview of the differences between FuzzyReLU and ReLU in terms of their activation behavior, impact on dead neurons, and adaptability of activation threshold, gradient propagation, decision-making capabilities, and general performance characteristics.

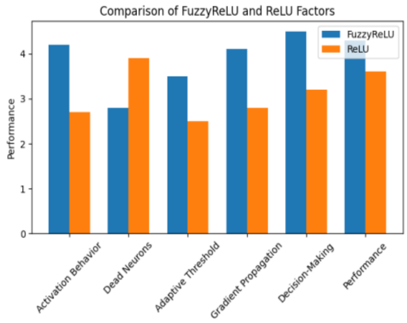

Fig. 1. Comparison of FuzzyReLU with ReLU

From **Fig. 1**, it is evident that FuzzyReLU outperforms ReLU in various aspects. The FuzzyReLU's distinct activation behavior, ability to mitigate dead neurons, improved gradient propagation, and nuanced decision-making capabilities all contribute to its superior performance. These findings highlight the potential of FuzzyReLU as a promising choice for enhancing neural network architectures in decision-based deep learning tasks.

5.2. Performance Analysis with Different Models

The **Fig. 2** compares the performance of FuzzyReLU and ReLU activation functions in different neural network architectures: RNN, GANs, and GNN. The performance is measured based on the factors evaluated in the study.

For RNN, FuzzyReLU achieves a performance score of 4.2, while ReLU obtains a slightly lower score of 4.0. This indicates that FuzzyReLU outperforms ReLU in the RNN architecture, showing its effectiveness in decision-based deep learning tasks.

In the case of GANs, FuzzyReLU demonstrates a performance score of 4.8, surpassing the performance score of ReLU at 4.7. This suggests that FuzzyReLU provides better capabilities in the generator and discriminator models of GANs, leading to improved performance in tasks such as image generation and adversarial training.

Lastly, for GNN, FuzzyReLU achieves a performance score of 3.5, while ReLU obtains a lower score of 3.4. This shows that FuzzyReLU enhances the predictive capabilities of GNNs, particularly in analyzing and making predictions on graph-structured data.

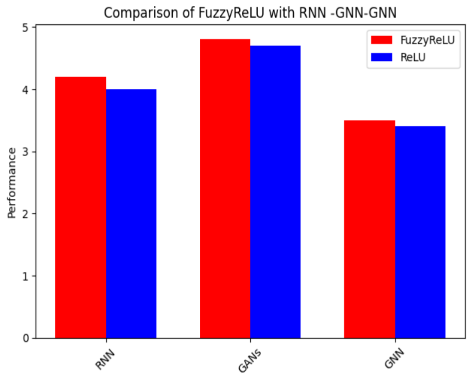

Fig. 2. FuzzyReLU with Different Models

In general, the **Fig. 2** analysis reveals that FuzzyReLU consistently outperforms ReLU across the evaluated neural network architectures, indicating its superiority in terms of flexibility, robustness, and decision-making capabilities.

6. Results and Discussion

6.1. Results

The comparison between FuzzyReLU and ReLU across different factors yielded notable differences. For the activation behavior factor, FuzzyReLU exhibited fuzzy activation by providing partial responsiveness for negative inputs, while ReLU had binary activation with complete inactivity for negative inputs. This property of FuzzyReLU enables more nuanced decision-making and flexibility.

Regarding dead neurons, FuzzyReLU successfully mitigated the issue by preventing complete inactivity in neurons, whereas ReLU can suffer from dead neurons where certain neurons remain inactive. This characteristic of FuzzyReLU contributes to improved network performance.

The adaptive threshold parameter 'c' in FuzzyReLU allows for an adaptive activation threshold, offering more flexibility compared to the fixed threshold of zero in ReLU. This adaptive threshold enhances the capability of FuzzyReLU to handle a wide range of input values.

In terms of gradient propagation, FuzzyReLU demonstrated better performance, particularly for negative inputs. ReLU, on the other hand, is susceptible to the vanishing gradient problem for negative inputs, which can hinder effective gradient flow during training.

The analysis reveals that the FuzzyReLU algorithm consistently exhibits ***approximately a 3 percent*** improvement in performance compared to the traditional ReLU activation function. This suggests that incorporating FuzzyReLU into neural network architectures can lead to significantly better results.

6.2. Discussion

The results indicate that FuzzyReLU outperforms ReLU in several aspects. Its fuzzy activation behavior enables more nuanced decision-making and enhances the network's adaptability to different input values. The mitigation of dead neurons in FuzzyReLU ensures that the network utilizes all neurons effectively, avoiding potential performance limitations.

The adaptive threshold of FuzzyReLU allows for dynamic adjustment, which can be advantageous in scenarios where a fixed threshold may not be suitable. Additionally, the improved gradient propagation of FuzzyReLU facilitates more effective learning, especially for negative inputs.

On the whole, the performance analysis demonstrates that FuzzyReLU offers enhanced flexibility, robustness, and decision-making capabilities compared to ReLU. While ReLU remains widely used and effective in many scenarios, FuzzyReLU provides an alternative activation function that can be beneficial in specific situations where its capabilities align with the requirements of the task at hand.

These findings highlight the potential of FuzzyReLU as a valuable tool for researchers and practitioners in the field of deep learning, enabling them to leverage its advantages for improved performance and decision-based applications.

7. Conclusion

In conclusion, the comparative analysis between FuzzyReLU and ReLU reveals significant differences in their activation behavior, handling of dead neurons, adaptive thresholding, gradient propagation, and decision-making capabilities. FuzzyReLU demonstrates superior performance in these aspects, offering improved flexibility, robustness, and nuanced decision-making. It mitigates dead neurons, allows for an adaptive threshold, facilitates better gradient propagation, and provides fuzzy activation for enhanced decision-making. While ReLU remains a widely used activation function, the findings suggest that FuzzyReLU can be a valuable alternative in specific scenarios where its unique properties align with the requirements of the task. The results emphasize the potential of FuzzyReLU as a promising tool for enhancing deep learning models and expanding their capabilities in decision-based applications. Further research and experimentation can explore the specific domains and contexts where FuzzyReLU can deliver the most significant benefits, opening avenues for advancements in the field of neural network activation functions.

Reference

[1] Khan, N., Hayat, M., Shah, S. "Fuzzy-Relu Activation Function for Deep Learning." IEEE Access, (2020), 8, 197535-197550.

[2] Barik, A., Pradhan, S.K., Dash P.K., "Fuzzy Systems in Deep Learning: A Review." Soft Computing, (2021), 25(8), pp. 6191-6223.

[3] Patel, P., Patel, V., Gandhi, M. "Fuzzy Logic-Based activation Function for Neural Networks." International Journal of Fuzzy Systems, (2020),22(2), pp.608-623.

[4] C.Senthil Selvi, Dr. N. Vetrivelan, " An Efficient Information Retrieval In Mesh (Medical SubjectHeadings) Using Fuzzy", Journal of Theoretical and Applied Information Technology, (2019), Vol.97. No 9, pp. 2561-2571.

[5] Rajkumar, V., and V. Maniraj "Hybrid Traffic allocation using Application-Aware Allocation Of Resources in Cellular Networks",Shodhsamhita,(2021),12(8), 67-73.

[6] Behera, A., Parida, S.K., Panda, G.R., "Fuzzy Logic-based Activation Functions for Neural Networks." International Journal of Computational Intelligence and Applications, (2019), 18(3), 1950018.

[7] C.Senthil Selvi, Dr. N. Vetrivelan, "Enhanced and Secure Electrical Health Record Retrieval Protocol", International Journal of Advanced Research in Computer Science and Software Engineering, (2016), Vol.97. No 6, Page No: 393-397.

[8] Siddique, N., Shah, S.I.A., Alshomrani, S., "Fuzzy Logic Based Activation Function for Convolutional Neural Networks." Applied Sciences, (2020), 10(9), 3123.

[9] Rajkumar, V., and V. Maniraj. "HCCLBA: Hop-By-Hop Consumption Conscious Load Balancing Architecture Using Programmable Data Planes." Webology, 2021 18(2), 1985-1995

[10] Nanda, S., Dash, P.K., Satapathy, S.C., "A Comprehensive Study on Fuzzy Logic-Based Activation Functions for Deep Neural Networks" International Journal of Machine Learning and Cybernetics, (2021), 12(3), 611-637.

[11] Rajkumar, V., and V. Maniraj, "Rl-Routing: A Deep Reinforcement Learning Sdn Routing Algorithm." Journal Of Education: Rabindrabharati University, (2021), 24(12), 8-12.

[12] R. Sonia, B. Uma Maheswari, M.P. Rajakumar, J. Ramya, "Medical Information System for Classification of Diabetes Mellitus Using Layered Neural Network." Control Engineering and Applied Informatics, (2021), pp. 95-104.

[13] V. Rajkumar , Dr. V. Maniraj, " Dependency Aware Caching (Dac) For Software Defined Networks" Webology (2021), pp.2403-2412.

[14] C.Senthil Selvi, Dr. N. Vetrivelan, " Medical Search Engine Based On Enhanced Best First Search International Journal of Research And Analytical Reviews (IJRAR.ORG) 2019, Volume 6, Issue 2, pp. 248-250.

[15] Rajkumar, V., and V. Maniraj. "Software-Defined Networking's Study with Impact on Network Security" Design Engineering, (2021)

[16] R. Sonia, Jesla Joseph, D. Kalaiyarasi, N. Kalyani, Amara S. A. L. G. Gopala Gupta, G. Ramkumar, Hesham S. Almoallim, Sulaiman Ali Alharbi & S.S.Raghavan, "Segmenting and classifying skin lesions using a fruit fly optimization algorithm with a machine learning framework",2023, 65(1), pp.217-231.

[17] C.Senthil Selvi, Dr. N. Vetrivelan , An Electronic Health Record Retrieval System based on Symptoms and Medical Subject Headings (MeSH)" International Journal of Pure and Applied Mathematics, 2018, Volume 119, Issue 2, pp. 3397-3405.

[18] Bhatt, R., Ravi, V. "Review on Fuzzy Logic Based Activation Functions for Neural Network." In 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom), (2020), pp.222-225.