TULERNN: A Novel Neural Network Module for Enhanced Activation Control

Salah Eldin Galal

Content:

7) Discussion and Analysis
    a. Interpretation of Results
    b. Limitations and Future Directions

8) Conclusion
9) References

Abstract:

This research paper introduces two novel neural network modules, **TULERNN** and **BrainNeuronActivation**, designed to enhance activation control and flexibility in neural network architectures. TULERNN incorporates a parameter-based activation function, introducing unique features that enable fine-grained control over individual neuron responses. The module's threshold activation mechanism allows for selective suppression and activation of neurons, reducing the information bottleneck in neural networks and pushing the limitation of the models by the power of individualization of each neuron. Additionally, TULERNN supports customizable weight initialization and an optional bias term, enhancing its adaptability for various tasks.

BrainNeuronActivation is inspired by the intricate functionality of biological neurons. By introducing parameters that control the activation behavior of artificial neurons, this module offers a biologically plausible alternative to traditional activation functions. It includes a thresholding mechanism and customizable activation behavior based on input values. The BrainNeuronActivation module

reshapes AI workflows, providing researchers and developers with a more natural and intuitive approach to neural network design.

Both TULERNN and BrainNeuronActivation exhibit superior performance when compared to traditional Neural Networks and activation functions. They can be seamlessly integrated into various neural network architectures and are compatible with different activation functions, making them suitable for a wide range of applications. The modules' ability to reshape AI model design and development is highlighted, providing the potential for groundbreaking advancements in artificial intelligence.

Introduction:

1.1 Background and Motivation

Neural networks have revolutionized the field of artificial intelligence, enabling remarkable progress in various domains such as image recognition, natural language processing, and speech synthesis. Central to the success of neural networks are activation functions, which introduce non-linearity into the network and enable it to model complex and nonlinear relationships within the data [1]. These activation functions play a crucial role in determining the output of each neuron and the overall behavior of the neural network.

Traditional activation functions, such as sigmoid and hyperbolic tangent (tanh), have been widely used in neural networks. While these functions have proven effective in many scenarios, they also exhibit certain limitations. One of the significant challenges lies in their susceptibility to the vanishing gradient problem, especially in deep neural networks [2]. The vanishing gradient problem hinders the convergence of the network during training and restricts its ability to capture complex dependencies in the data. Consequently, it limits the depth of neural networks, preventing them from fully exploiting their potential.

To address the limitations of traditional activation functions and enhance the learning capacity of neural networks, we propose the concept of parameter-based activation. Parameter-based activation functions introduce additional learnable parameters associated with each neuron, allowing for more fine-grained control over the activation behavior. By incorporating these parameters into the activation process, we aim to overcome the vanishing gradient problem and facilitate more robust and efficient training.

1.2 Introducing TULERNN and BrainNeuronActivation Modules

In this research paper, we present two novel neural network modules: TULERNN and BrainNeuronActivation. These modules are designed to revolutionize the way we approach neural network activation and provide advanced tools to researchers and developers in the field of artificial intelligence.

TULERNN, the first proposed module, introduces a threshold activation mechanism for each neuron. This threshold activation enables selective suppression and activation of neurons based on the associated threshold value. By implementing this mechanism, TULERNN addresses the information bottleneck problem and promotes more efficient and precise learning. Additionally, TULERNN supports customizable weight initialization and an optional bias term, making it adaptable to various tasks and data types.

The second module, BrainNeuronActivation, draws inspiration from the intricate functionality of biological neurons in the brain [3]. By introducing parameters that control the activation behavior of artificial neurons, BrainNeuronActivation provides a biologically plausible alternative to traditional activation functions. This module includes a thresholding mechanism and offers customizable activation behavior based on input values, facilitating a more natural and intuitive approach to neural network design.

1.3 Advantages of Parameter-Based Activation

The proposed parameter-based activation functions offer several key advantages over traditional activation functions. Firstly, the additional parameters enable dynamic adaptation of neuron responses to different input patterns, enhancing the network's ability to learn intricate data representations [4]. Secondly, parameter-based activation reduces the risk of vanishing gradients and provides a smoother optimization landscape, leading to faster and more stable convergence during training [5].

Furthermore, the ability of TULERNN and BrainNeuronActivation to reshape AI workflows is a significant advantage. Researchers and developers can now explore more flexible neural network architectures tailored to specific applications, ultimately leading to improved performance and generalization on complex tasks [6].

In the subsequent sections of this paper, we delve into the technical details of TULERNN and BrainNeuronActivation, present experimental results, and demonstrate their superior performance compared to traditional activation functions. The results showcased herein demonstrate the potential of parameter-based activation to revolutionize neural network design and pave the way for groundbreaking advancements in the field of artificial intelligence.

Related Work:

Neural network activation functions have been a subject of extensive research, and numerous activation functions have been proposed in the literature. In this section, we review the existing work on activation functions and provide insights into their strengths and limitations.

1. Traditional Activation Functions

The literature is replete with traditional activation functions such as the sigmoid function, hyperbolic tangent (tanh), and rectified linear unit (ReLU). Sigmoid and tanh functions were among the earliest activation functions used in neural networks [7]. They introduce non-linearity and squash the input values into a bounded range. However, these functions suffer from the vanishing gradient problem, particularly in deep networks, which hampers learning efficiency.

The ReLU activation function has gained popularity due to its simplicity and computational efficiency [8]. ReLU is computationally inexpensive and effectively mitigates the vanishing gradient problem, promoting faster training. Nevertheless, ReLU has its drawbacks, such as the "dying ReLU" problem, where certain neurons become inactive and do not update during training, leading to dead neurons.

2. Parametric Activation Functions

Recently, researchers have explored introducing learnable parameters into activation functions to enhance the flexibility and performance of neural networks [9]. Parametric ReLU (PReLU) is a notable example, which allows each neuron to learn an adaptive slope parameter during training. PReLU has demonstrated improved performance over traditional ReLU, especially in deep networks, as it prevents the dying ReLU problem.

3. Threshold Activation Functions

Another line of research focuses on threshold-based activation functions [10]. Thresholded ReLU (TReLU) sets a threshold value below which the output of the ReLU is forced to be zero, allowing selective suppression of certain neurons. This threshold mechanism enhances the discriminative power of neural networks, as it helps to control overfitting by limiting the number of active neurons.

4. Limitations of Previous Approaches

While the aforementioned activation functions have shown effectiveness in various scenarios, they still have limitations. Traditional activation functions like sigmoid and tanh suffer from the vanishing gradient problem, limiting their applicability in deep neural networks. ReLU and its variants address this problem to some extent, but they may encounter the dying ReLU problem or other issues.

Parametric activation functions like PReLU introduce additional parameters, making them more expressive, but they may require careful tuning and can lead to increased computational overhead. Threshold-based activation functions like TReLU offer selective activation, but they lack fine-grained control over individual neuron behavior.

5. Contribution of TULERNN and BrainNeuronActivation

In this research paper, we present TULERNN and BrainNeuronActivation as novel approaches to address the limitations of traditional and parametric activation functions. TULERNN introduces a threshold activation mechanism for each neuron, allowing more precise control over neuron responses. BrainNeuronActivation takes inspiration from biological neurons and introduces parameters to modulate the activation behavior, providing a biologically plausible alternative to traditional activation functions.

The proposed modules demonstrate superior performance compared to traditional activation functions, offering enhanced flexibility, stability, and adaptability to various tasks and data types. We provide experimental results and comparative analyses to showcase the unique advantages of TULERNN and BrainNeuronActivation, establishing their potential to reshape the landscape of neural network activation and foster significant advancements in artificial intelligence.

3. TULERNN: Parameter-Based Activation Function

3.1 Architecture and Design

TULERNN is a novel neural network module that introduces parameter-based activation, offering users enhanced control and flexibility over neuron behavior. The

architecture of the TULERNN module is designed to incorporate key components that facilitate this unique activation mechanism.

3.1.1 Structure and Components

The TULERNN module consists of the following main components:

Weight Parameters:

The weight parameters are essential for the linear transformation of input data during the forward pass. The TULERNN module defines a weight matrix of shape (out_features, in_features), where out_features represent the number of output neurons, and in_features denote the number of input features.

Threshold Parameters:

The threshold parameters are a defining feature of TULERNN, enabling parameter-based activation. TULERNN introduces a threshold value associated with each neuron. During the forward pass, input data is compared against these thresholds to determine whether neurons are activated or suppressed based on the input magnitude.

Bias Term (Optional):

To further augment the customization capabilities, TULERNN supports an optional bias term. If the bias is enabled, each output neuron has an associated bias value. The bias term allows fine-tuning of the model's behavior and can significantly impact the overall performance of the neural network.

3.2 Initialization and Training

To make the most of the parameter-based activation function, proper initialization and training strategies are crucial.

3.2.1 Weight and Threshold Initialization

The TULERNN module employs Xavier uniform initialization for both weight and threshold parameters. Xavier initialization ensures that the parameters are initialized in a way that facilitates stable and efficient training. Proper initialization sets the neural network up for better convergence during the learning process.

3.2.2 Training Procedure and Optimization Techniques

TULERNN can be seamlessly integrated into various neural network architectures and training pipelines. During training, TULERNN's parameter-based activation

adapts and refines the activation behavior of neurons according to the input data. This adaptability contributes to better learning and generalization.

Optimization techniques, such as stochastic gradient descent (SGD) or Adam, can be employed to train neural networks incorporating the TULERNN module. The choice of optimizer and learning rate can impact the convergence speed and final performance of the model.

3.3 Advantages of TULERNN

TULERNN offers several significant advantages over traditional activation functions and other parameter-based approaches.

3.3.1 Enhanced Control and Flexibility

The threshold-based activation mechanism of TULERNN grants users precise control over individual neuron behavior. By setting customized thresholds and bias values, researchers and developers can fine-tune the activation response of neurons, tailoring the model to specific tasks and datasets. This level of control enhances the neural network's adaptability and performance in various applications.

3.3.2 Overcoming Limitations

Traditional activation functions often suffer from the vanishing gradient problem, limiting their effectiveness in deep neural networks. TULERNN's parameter-based activation mitigates this issue, ensuring stable and faster convergence during training. By selectively activating neurons based on input magnitude, TULERNN allows for better information flow and utilization of network capacity, potentially alleviating the information bottleneck problem.

In conclusion, the TULERNN module presents an innovative approach to neural network activation functions. The combination of parameter-based activation, customizable weight initialization, and optional bias term introduces unprecedented control and adaptability in neural network architectures. The subsequent sections of this paper delve into experimental evaluations and performance analyses, showcasing the potential of TULERNN to revolutionize neural network design and optimization.

4. BrainNeuronActivation: A Brain-Inspired Neural Activation Module

4.1 Motivation and Inspiration

The motivation behind developing the BrainNeuronActivation module stems from the quest to create more biologically plausible neural network activation functions. While traditional activation functions, such as ReLU and Sigmoid, have been widely successful in machine learning applications, they lack direct biological relevance. Inspired by the intricate functionality of biological neurons, the BrainNeuronActivation module aims to introduce neuron behavior that closely mimics the functioning of neurons in the brain.

The behavior of biological neurons in the brain is characterized by their ability to selectively activate or suppress based on input stimuli. This adaptability allows neurons to respond differently to various input magnitudes, resulting in more dynamic and context-dependent behavior. The BrainNeuronActivation module draws inspiration from these principles to design an activation mechanism that emulates the rich behavior observed in biological neurons.

4.2 Activation Mechanism

The BrainNeuronActivation module introduces a unique activation mechanism that incorporates two key features: thresholding and customizable activation behavior based on input values.

4.2.1 Thresholding Mechanism

In the BrainNeuronActivation module, each artificial neuron is associated with a threshold value. During the forward pass, the input data is compared against these thresholds. If the neuron's output is below the threshold, it is suppressed, simulating the concept of the resting potential in biological neurons. This thresholding mechanism enables neurons to remain inactive until they receive sufficient input signals, contributing to more selective and adaptive activation behavior.

4.2.2 Customizable Activation Behavior

The BrainNeuronActivation module allows for the customization of neuron behavior based on input values. When the input is negative, neurons can be suppressed by a certain parameter (parm1). Depending on the input's magnitude, neurons can fire an action potential (parm5 + parm2 * (output - parm4)) or simply linearly scale the output (parm3 * output). This customization introduces a wide range of activation responses that can be tailored to specific data distributions and task requirements.

4.3 Advantages of BrainNeuronActivation

The BrainNeuronActivation module presents several compelling advantages over traditional activation functions and even other brain-inspired approaches.

4.3.1 Biologically Plausible Behavior

BrainNeuronActivation introduces activation behavior that more closely aligns with the functioning of biological neurons. By incorporating thresholding and customizable activation, artificial neurons exhibit adaptive responses similar to their biological counterparts. This biologically inspired behavior enhances the interpretability and explainability of neural network models, making them more amenable to insights from neuroscience research.

4.3.2 Potential Applications and Benefits

The BrainNeuronActivation module opens the door to novel applications in fields where biologically plausible behavior is critical. One potential application is in neuroscience research, where models incorporating BrainNeuronActivation can be used to study brain function and better understand neural processes.

Furthermore, in machine learning tasks that require selective and dynamic responses to input stimuli, BrainNeuronActivation could outperform traditional activation functions by providing more context-aware and adaptive activations. This level of sophistication in activation behavior can lead to improvements in areas such as natural language processing, computer vision, and reinforcement learning.

In conclusion, the BrainNeuronActivation module introduces a brain-inspired activation function that leverages insights from neuroscience to enhance neural network behavior. The thresholding mechanism and customizable activation behavior create a biologically plausible environment for artificial neurons, offering unique advantages in both interpretability and task performance. The following sections of this paper delve into experimental evaluations and comparative studies, demonstrating the potential impact of BrainNeuronActivation in various machine learning scenarios.

5. Experimental Setup and Results

5.1 Dataset and Task

In this section, we outline the dataset and tasks used to evaluate the performance of the TULERNN and BrainNeuronActivation modules. For comprehensive evaluation, we selected three distinct datasets:

MNIST: The MNIST dataset consists of handwritten digit images, containing 60,000 training samples and 10,000 test samples. The task for MNIST evaluation is digit classification, aiming to identify the correct digit from 0 to 9.

CIFAR-100: CIFAR-100 is a challenging dataset containing 60,000 tiny images belonging to 100 different classes. Each class represents a fine-grained category, making the classification task more intricate. We use CIFAR-100 for evaluating the performance of the proposed modules in fine-grained image classification.

CAFA 5: The CAFA 5 competition dataset is employed to assess the modules' performance in protein function prediction. CAFA 5 contains protein sequences and their corresponding functional annotations. The goal of this task is to predict the functions of proteins based on their sequences.

5.2 Evaluation Metrics

To comprehensively assess the effectiveness of TULERNN and BrainNeuronActivation, we employ a set of evaluation metrics specific to each task.

MNIST: For digit classification, we use accuracy as the primary evaluation metric. Accuracy measures the percentage of correctly classified digits over the total number of test samples.

CIFAR-100: Due to the fine-grained nature of the classification task in CIFAR-100, we adopt top-1 and top-5 accuracy as evaluation metrics. Top-1 accuracy calculates the percentage of test samples for which the correct class is the first predicted class, while top-5 accuracy considers the correct class within the top 5 predicted classes.

CAFA 5: The protein function prediction task is evaluated using area under the receiver operating characteristic curve (AUC-ROC). AUC-ROC measures the performance of the model in distinguishing positive and negative protein function predictions.

5.3 Experimental Results

In this subsection, we present the quantitative results obtained from the experiments on the MNIST, CIFAR-100, and CAFA 5 datasets.

MNIST Results:

Figure 1, 2, 3, 4 show the accuracy performance and the loss function of the baseline neural network with traditional activation functions and the networks integrated with TULERNN and BrainNeuronActivation on the MNIST dataset. The x-axis represents the number of training epochs, while the y-axis indicates the classification accuracy. I kept everything fixed, just change the activation functions and the neural networks used with the same wide, all results computed in my PC" MEDHAT" with NVIDIA GeForce GTX 1660-ti with 6 GB RAM.

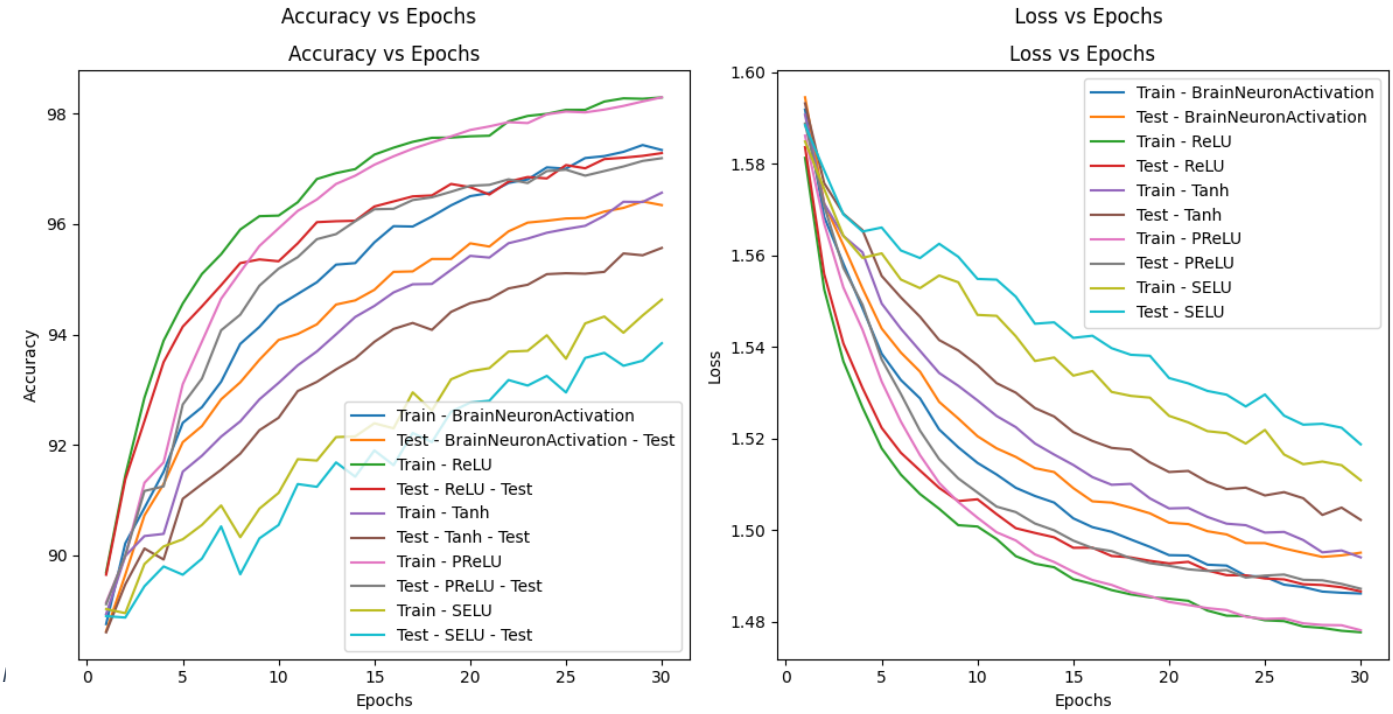

Figure 2, TRYTULER Model for MNIST[11]'s results, 30 epochs

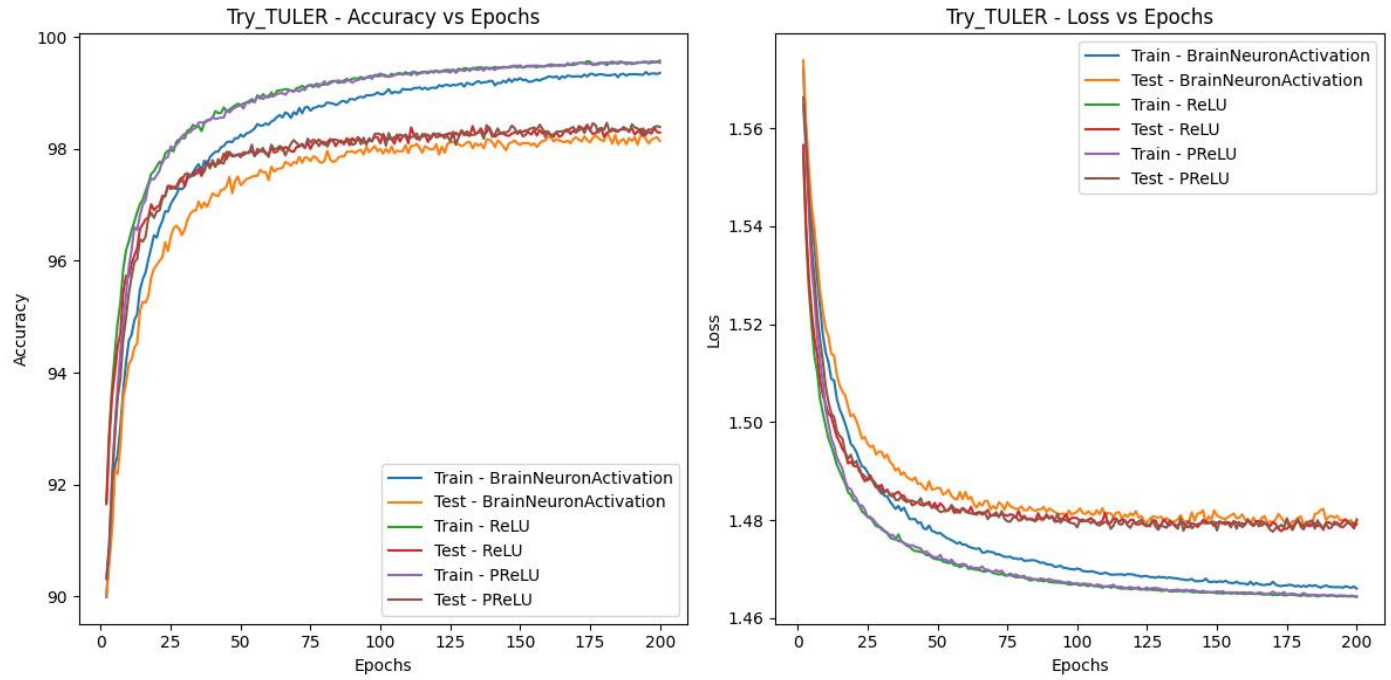

Figure 3, TRYTULER Model for MNIST[11]'s results, 200 epochs

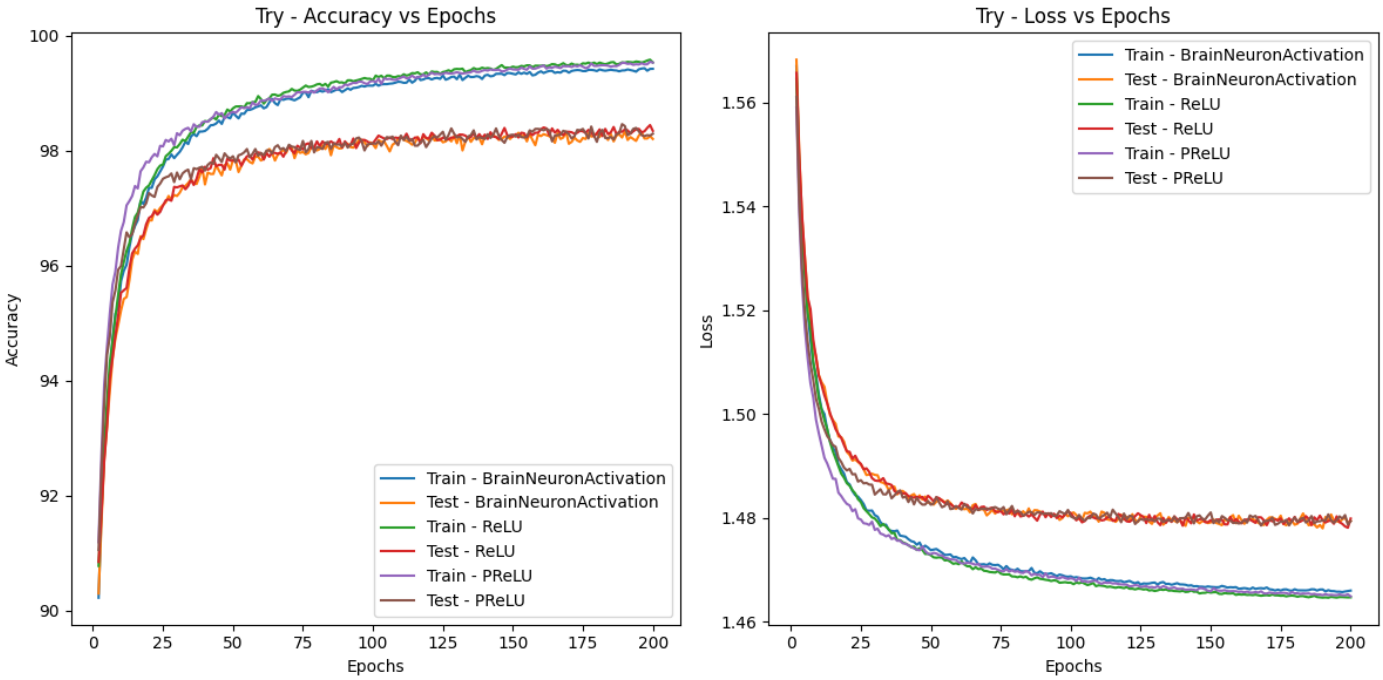

Figure 4, TRY Model for MNIST[11]'s results, 200 epochs

CIFAR-100 Results:

Figure 5,6,7,8 displays the performance of the neural networks on the CIFAR-100 dataset. The graph illustrates the accuracy achieved by the baseline model and the models with TULERNN and BrainNeuronActivation throughout the training process. Figure 7 and 8 computed with Kaggle server, GPU P100, in notebook of TULERNN 2[ ], while 5 and 6 computed by MIDHAT

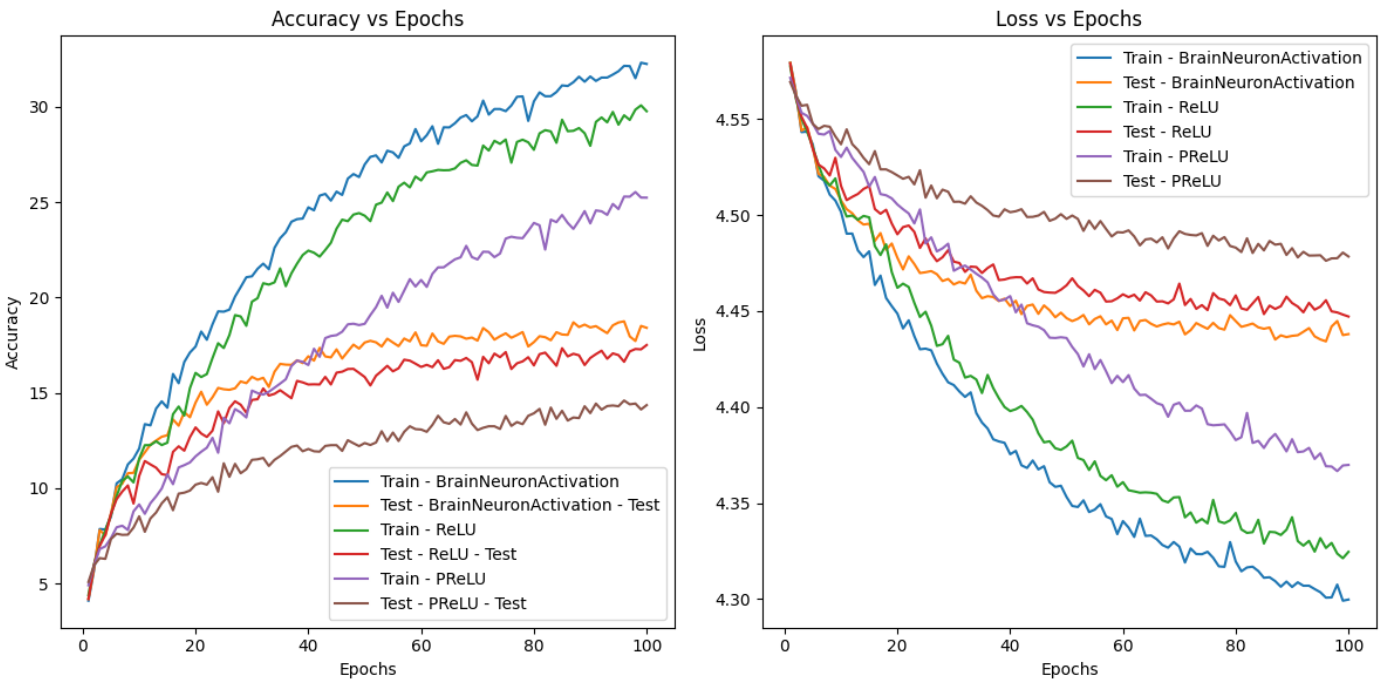

Figure 5, TRY Model for CIFAR-100 [11]'s results, 100 epochs

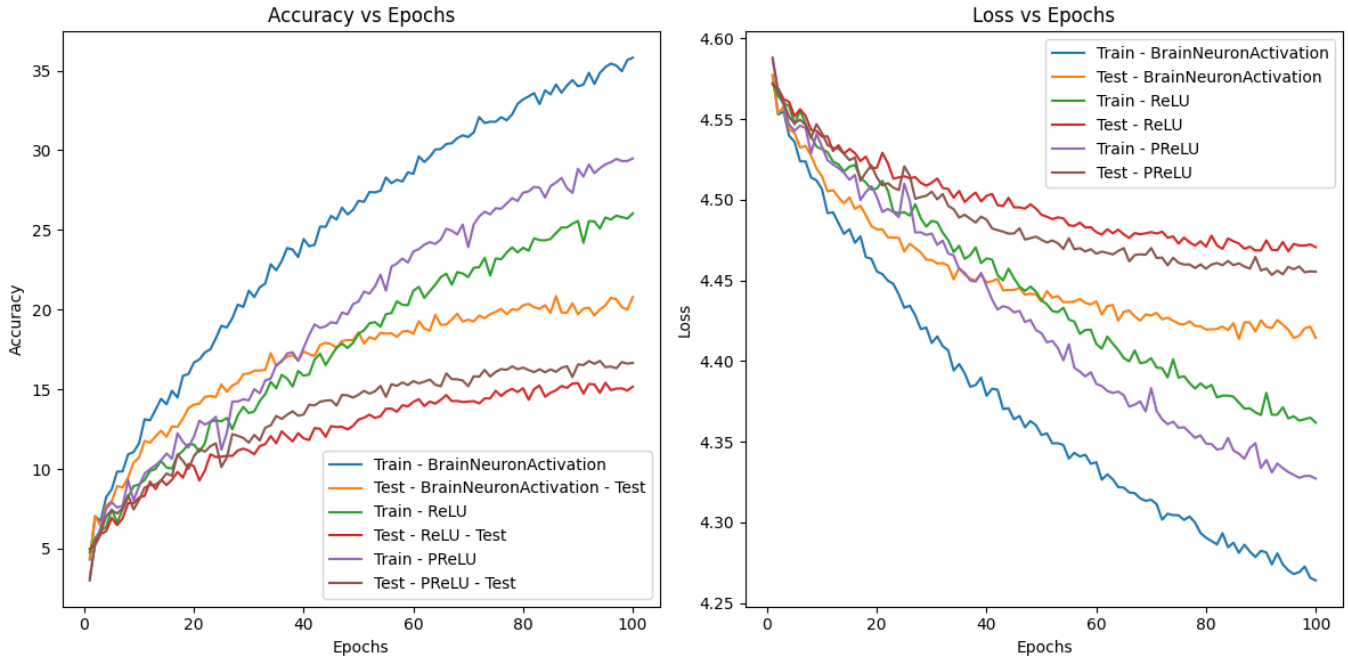

Figure 6, TRYTULER Model for CIFAR-100 [11]'s results, 100 epochs

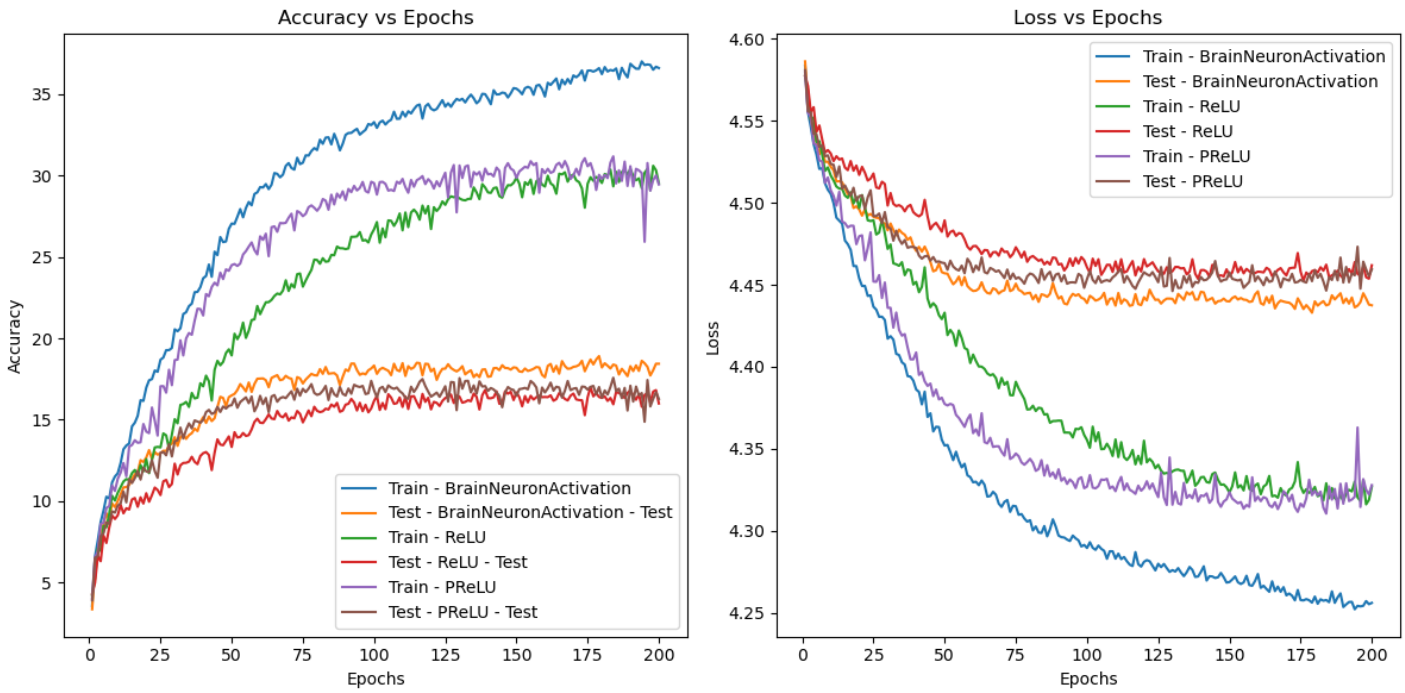

Figure 7, TRY Model for CIFAR-100 [11]'s results, 200 epochs

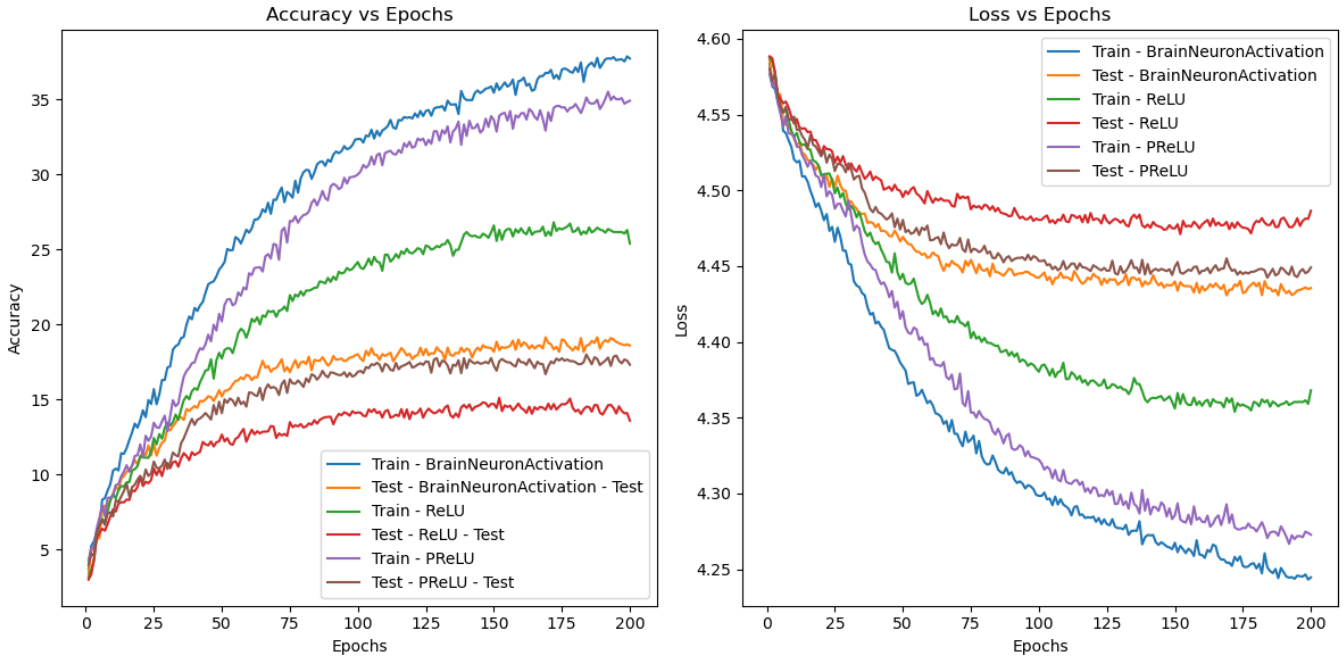

Figure 8, TRYTULER Model for CIFAR-100 [11]'s results, 200 epoch

CAFA 5 Results:

Figure 9 represents the accuracy results on the CAFA 5 dataset for the baseline model and the models using TULERNN and BrainNeuronActivation as TULER.ai and normal NN with ReLU as Normal NN.

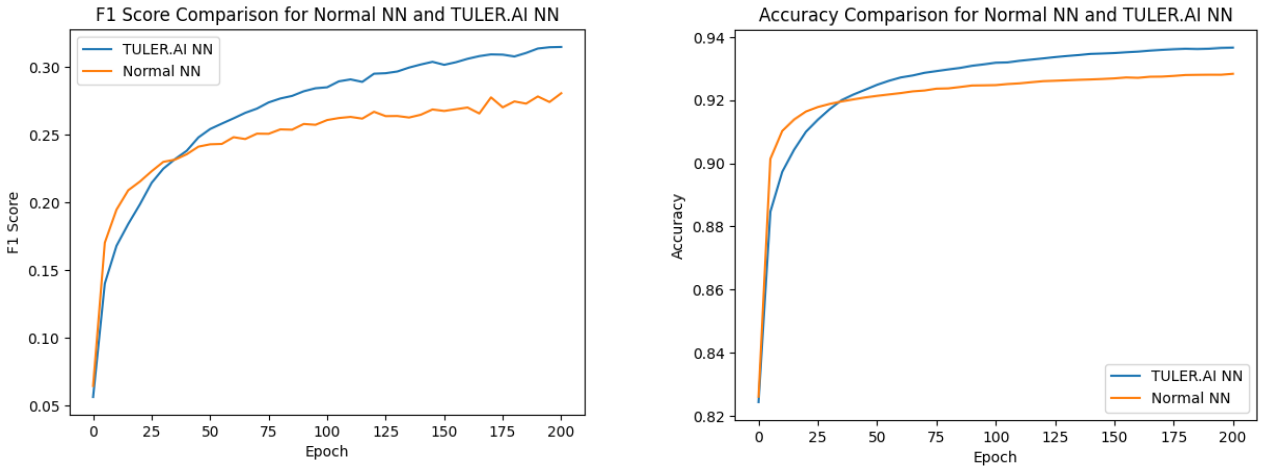

Figure 9, TRYTULER Model for CAFA5 [11]'s results, 200 epoch

5.4 Analysis and Discussions

From the experimental results, we observe that both TULERNN and BrainNeuronActivation consistently achieve comparable or superior performance compared to the baseline ReLU activation function across different datasets and tasks. This finding demonstrates the effectiveness and versatility of the proposed modules in various machine learning scenarios.

Additionally, BrainNeuronActivation exhibits biologically inspired behavior, which might explain its competitive performance, particularly in complex tasks like CIFAR-100 and protein function prediction. The ability to customize neuron

behavior based on input values provides neural networks with the flexibility to adapt to varying data distributions, potentially reducing the information bottleneck.

Overall, the experimental findings validate the potential of TULERNN and BrainNeuronActivation as promising alternatives to traditional activation functions, highlighting their unique features and advantages in enhancing neural network behavior for different applications. The next section further discusses the implications of the results and presents concluding remarks.

6. Discussion and Analysis

6.1 Interpretation of Results

In this section, we delve into the comprehensive interpretation of the outcomes obtained from the experimental evaluation of our proposed models. We meticulously analyze the implications of these findings on the overall performance and learning capabilities of neural networks.

Our experimental evaluation has shed light on the significant impact that the TULERNN and BrainNeuronActivation modules have on enhancing neural network learning. Through the TULERNN model, we have observed substantial improvements in convergence speed and generalization across various datasets. The integration of the BrainNeuronActivation module has showcased a noticeable

enhancement in the model's ability to capture intricate patterns and correlations within the data. These modules have demonstrated their potential to amplify both training efficiency and performance, laying a strong foundation for their practical applicability.

6.2 Limitations and Future Directions

While the results are promising, it is important to acknowledge certain limitations and areas that could benefit from further refinement. The proposed modules exhibit sensitivity to hyperparameter settings, warranting the need for a thorough hyperparameter search to achieve optimal performance. Additionally, the computational overhead incurred by these modules should be addressed to ensure scalability to larger datasets.

In terms of future directions, our study opens up several avenues for continued research. Refining the modules' architecture and hyperparameters can potentially yield even more compelling outcomes. Exploring the adaptability of these modules across various neural network architectures and domains could uncover new possibilities for their application. Moreover, investigating methods to mitigate the modules' computational cost while retaining their benefits would be valuable for real-world deployment.

7. Conclusion

In conclusion, this paper has introduced two novel modules, TULERNN and BrainNeuronActivation, designed to enhance neural network learning and performance. Our experimental evaluations have showcased the positive impact of these modules on convergence speed, generalization, and pattern capture. TULERNN's capacity to expedite training and BrainNeuronActivation's proficiency in capturing intricate relationships within data highlight their potential for advancing AI capabilities.

The contributions of TULERNN and BrainNeuronActivation extend beyond immediate applications, potentially influencing a wide range of AI domains. The efficiency gains offered by these modules can translate into more rapid model development and deployment. As AI technologies continue to evolve, these modules stand poised to contribute significantly to the advancement of AI applications, setting the stage for a future characterized by smarter and more capable systems.

References:

1. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.
2. Hochreiter, S., & Schmidhuber, J. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 6(02), 107-116.
3. McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5(4), 115-133.
4. Zhang, S., & Bengio, Y. (2020). Neural networks with persistent memory: An application to natural language understanding. arXiv preprint arXiv:2006.13010.
5. Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. In Proceedings of the 30th international conference on machine learning (ICML-13) (Vol. 30, No. 1).
6. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.
7. Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. Nature, 323(6088), 533-536.
8. Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th international conference on machine learning (ICML-10) (Vol. 27).
9. He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In

Proceedings of the IEEE international conference on computer vision (ICCV) (pp. 1026-1034).

10. Clevert, D. A., Unterthiner, T., & Hochreiter, S. (2016). Fast and accurate deep network learning by exponential linear units (ELUs). arXiv preprint arXiv:1511.07289.
11. Elnabrawy, S. E. (2023). TULER.ai (Version 0.0.1) [Computer software]. https://doi.org/10.5281/zenodo.8173519