Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 5, No. 1, 2023

A Chaotic Neuron and its Ability to Prevent Overfitting

Xiu Chen, Yi Wang *

Dept of Cyber space Security Academy, Chengdu University of Information Technology, Chengdu, 610225, China

Abstract: Chaotic neuron is a neural model based on chaos theory, which combines the complex dynamic behavior of biological neurons with the characteristics of chaotic systems. Inspired by the chaotic firing characteristics of biological neurons, a novel chaotic neuron model and its response activation function LMCU are proposed in this paper. Based on one-dimensional chaotic mapping, this chaotic neuron model takes the emissivity of chaotic firing characteristics of biological neurons as its response output, so that it has the nonlinear response and chaotic characteristics of biological neurons. Different from the traditional neuron model, it makes full use of the nonlinear dynamics of the chaotic system to achieve the activation output. In this paper, we apply the proposed chaotic neurons to artificial neural networks by using LeNet-5 models on MNIST and CIFAR-10 datasets, and compare them with common activation functions. The application of chaotic neurons can effectively reduce the overfitting phenomenon of artificial neural network, significantly reduce the generalization error of the model, and greatly improve the overall performance of artificial neural network. The innovative design of this chaotic neuron model provides a new cornerstone for the future development of artificial neural networks.

Keywords: Chaos; Activation Function; Overfitting; Biological Neuron; Neural Network.

1. Introduction

Artificial neural network [1-2] processes external data by imitating the connection structure and stimulus response mechanism of human brain, in which the activation function of artificial neurons acts on data to simulate the processing of electrical signals by biological neurons [3]. In biological neuronal response, activation function can realize nonlinear transformation, provide information transfer and processing capabilities, and have adjust ability. These characteristics enable the activation function to better simulate and approximate the response of biological neurons, but there is still a huge difference between the artificial neuron model and the biological neuron model. The former is a high simplification of the response characteristics of the latter and lacks the authenticity of biological response [4]. While one of the most interesting properties of the brain is its ability to display chaos [5], to take full advantage of the human brain's excellent learning capabilities, many researchers have designed biologically inspired algorithms and network architectures. The model based on chaotic neurons is a valuable neural network because its chaotic behavior has the characteristics of chaotic brain.

In a theoretical explanation of memory coding in the brain, Freeman, Kozma, and their team developed a chaotic model to explain the process of remembering smells [6,7,8]. Tsuda et al. also studied chaotic neural networks as generators of short-term memory and dynamic linkage of long-term memory [9,10]. Kaneko explored the dynamics of globally coupled chaotic graphs, showing that these networks [11,12]. In the study of classification problems, Harikrishnan et al proposed ChaosNet, a new artificial neural network architecture for classification tasks based on chaos, in 2019 [13]. In a follow-up study, Harikrishnan et al. extracted four "neural chaos" features from neurons in the ChaosNet structure [13], which were used to train a linear classifier for a support vector machine. This hybrid approach produces superior performance under a low training sample regime on synthetically generated and real-world datasets [14]. In the following experimental study, Sethi et al. emphasized the performance comparison between ChaosNet learning algorithm and classical ML algorithm in learning in limited training examples [15]. In the latest study, Remya et al. proposed for the first time that two extended ChaosNet's heterogeneous neuronal structures are one step closer to the brain [16]. However, in the study of classification problems, researchers only combine the primitive characteristics of chaotic firing of biological neurons with machine learning algorithms for experimental research. However, artificial neural network is prone to overfitting and poor generalization ability in practical classification problems because of various complex factors such as model parameter design and overtraining. Activation function is an important part of artificial neurons. Many researchers have proposed many improved schemes for activation function to improve the ability of preventing overfitting and model generalization in actual training process. In 2010, Hinton proposed relu [17] as an activation function for the first time. Later, LeakyReLU[18], ELU[19],Swish[20], Maxout[21], Mish[22] and other activation functions were proposed to improve the limitations of traditional ReLU functions. They have certain advantages and effects in preventing overfitting and improving model generalization ability. In China, some researchers have proposed Xwish[23], S-Relu[24], RELU-XEX[25] activation functions, which effectively reduce the "neuronal necrosis" of relu. Even though the newly proposed oscillatory activation function [26] has biological inspiration, these improved activation function curves are basically smooth, derivable, and obviously without spikes, which are only highly simplified responses of biological neurons, lacking the chaotic response characteristics of biological neurons.

Therefore, inspired by the chaotic discharge characteristics of biological neurons, this paper proposes a novel chaotic neuron and its activation function with chaotic response characteristics of biological neurons. Based on one-dimensional chaotic mapping, this chaotic neuron adds chaotic characteristics to the response output process of artificial neurons, and takes the emissivity of chaotic discharge characteristics of biological neurons as its response

output. Different from the traditional neuron model, it uses the nonlinear dynamics of chaotic system to realize the activation response output. This transformation process realizes the chaotic response output of artificial neurons for the first time, which provides a new cornerstone for the development of artificial neural networks in the future. In this paper, it is applied to artificial neural networks. In comparison with other common activation functions, LeNet-5 model is used on data sets MNIST and CIFAR-10, and it is found that the use of chaotic neurons can effectively inhibit the overfitting of artificial neural networks and greatly reduce the generalization error of the model. The overall performance of artificial neural network is improved. Compared with the mainstream anti-overfitting methods, the proposed chaotic neuron and its activation function reduce the dependence of artificial neural networks on the mainstream anti-overfitting methods, thus improving the efficiency of model training.

2. Background

2.1. Biological Neurons

A biological neuron is the fundamental unit of the human brain. It receives input signals from other neurons through dendrites, processes the signals within its cell body, and transmits them through an axon. When the accumulated signal reaches a certain threshold, it generates an electric pulse to activate the output signal. The dynamic behavior of this process is highly complex and can exhibit various characteristics, including oscillations, synchronicity, chaos, and quasi-linearity. Quasi-linearity is one of the features in the response of biological neurons, representing a non-linear overall growth of neuron response within a certain range. When a biological neuron undergoes chaotic firing in response to signal stimulation, changes occur in ion channels both inside and outside the cell membrane, resulting in an increase in neuron activity levels. When the neuron activity level exceeds a threshold, it triggers the generation of an action potential. During the action potential, the flow of ions inside and outside the cell membrane leads to membrane potential repolarization. After the action potential, the neuron enters a refractory period, meaning it cannot generate new action potentials for a certain period of time.

Related concepts are involved in the chaotic firing process of biological neurons:

(1) Neuron activity level value: neuron activity level value is a quantitative representation of the current excitation state or activation degree of neurons.

(2) Threshold value: The threshold value is a specific neuronal activity level value, and when the activity level value exceeds the specific threshold activity level value, the neuron will be triggered to produce an action potential.

(3) Emissivity: Emissivity refers to the frequency at which a neuron generates an action potential over a period of time.

(4) Resting state: In the absence of external stimulation, the neuronal activity level value remains at a relatively stable value.

(5) Depolarization: Depolarization is the process of action potential, which gradually increases the level of neuronal activity. Usually involves the influx of sodium ions.

(6) Repolarization: After the neuron generates an action potential, the ion concentration inside and outside the cell membrane is read jugated, and the neuronal activity level gradually declines. Repolarization is a process of recovery, usually involving the outflow of potassium ions.

(7) Refractory period: Refractory period refers to the inability of neurons to generate new action potentials for a period of time, and the activity level value of neurons is difficult to increase again.

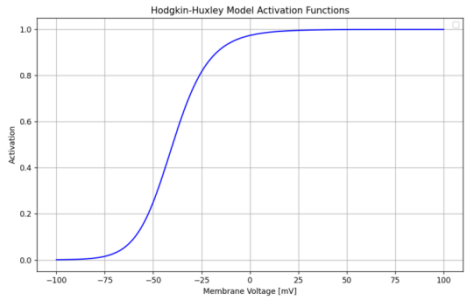


Figure 1. Sodium channel activation response diagram of Hodgkin-Huxley model (similar to sigmoid image)

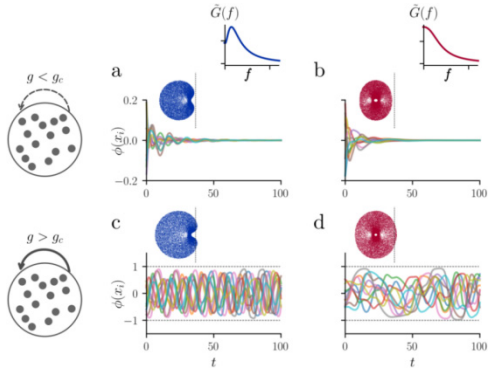
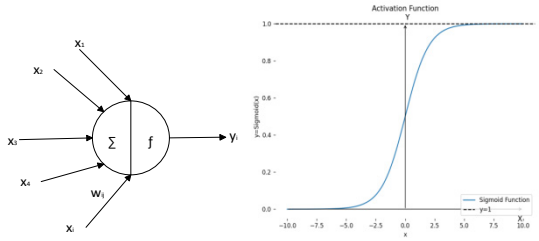


Figure 2. Response curve of chaotic discharge of biological neurons [28]

2.2. Artificial Neurons

Artificial neurons are an abstraction of the behavior of biological neurons, integrating all incoming signals at the synapse, and then processing the integrated signals using activation functions. Its processing process simulates the activation output of biological neurons. Figure3 illustrates the activation function of sigmoid, where x_i is the signal on the input synapse, w_{ij} is the strength of the synapse, f is the activation function, y_i represents the signal output value, b represents the bias.



$$y_i = f(\sum_{i=1}^{n} w_{ij} + b)$$

Figure 3. Schematic diagram of activation function

The activation function determines whether the artificial neuron will be activated, and it has nonlinear characteristics. Therefore, the introduction of activation function can enhance the nonlinear expression of the artificial neural network, which also enables the neural network to learn and approximate complex functions. Continuity and differentiability are ideal properties of the activation function, and they allow the use of gradient-based optimization methods to train neural networks so that artificial neural

networks can learn efficiently through techniques such as gradient descent. At the same time, an activation function should be monotonic, so that an increase in the input synaptic signal also leads to an increase in the output. It can be seen in Figure 3 that the sigmoid function generates outputs between 0 and 1, which is also a feature of the bounded output of the activation function. The bounded output can help prevent problems such as gradient explosion during training of the artificial neural network. Saturation behavior refers to the situation where the output of the activation function becomes very close to the minimum or maximum value, resulting in a very small gradient. Figure 3 shows the saturation behavior of sigmoid, in addition to the saturation behavior of tanh activation function.

2.3. One-dimensional Chaotic Mapping

Chaotic dynamics exist not only in individual neurons, but also in neural networks and the brain. The simplest dynamic system for displaying chaos is a one-dimensional chaotic map. One-dimensional chaotic mapping can be divided into Logistic Map [29] and Tent map [29]. Although one-dimensional chaotic maps are the simplest chaotic maps, they are still extremely sensitive to initial conditions, and small changes in initial conditions can lead to significantly different output trajectrajectory.

2.3.1. Logistic Map

Logistic Map generates a chaotic sequence in the range of [0,1] by scaling the initial values in the range of [0,1] [30]. It is mathematically defined as:

$$x_{n+1} = rx_n(1 - x_n) \tag{1}$$

r is the chaos control parameter, the value range is [0,4]. As can be seen from Figure 4., r when >3.57, a chaotic system appears, and the larger the chaos control parameter, the more chaotic the system.

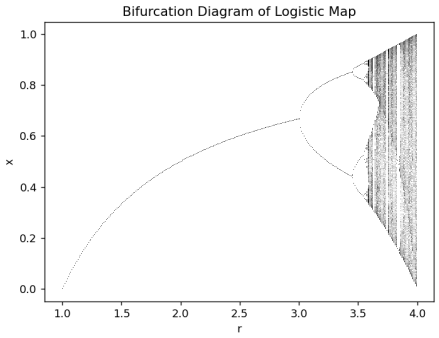


Figure 4. Logistic Map bifurcation diagram

2.3.2. Tent Map

Tent map is a piecewise linear mapping function. Because its mapping shape map is similar to a tent, it is also called tent mapping. Tent map structure is simple, and the map presented the result distribution density is relatively uniform, has a good ergodic. Its mathematical expression is:

$$x_{n+1} = \begin{cases} ux_n, x_n < 0.5 \\ u(1 - x_n), x_n \geq 0.5 \end{cases} \tag{2}$$

In the formula (2), the initial value x ranges from [0,1]. Is a chaos control parameter. The value ranges from 0 to 2. Figure 5 shows the bifurcation of Tent map. When the chaos control parameter u is greater than 1, chaos begins to appear in the system.

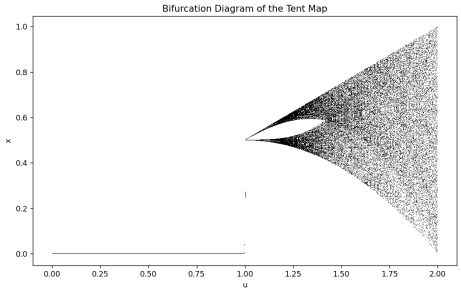


Figure 5. Tent Map bifurcation diagram

3. Chaotic Neurons

Chaotic neuron is a kind of neuron model based on chaos theory, which combines the complex dynamic behavior of biological neurons with the characteristics of chaotic system. The chaotic neuron proposed in this paper is based on one-dimensional chaotic mapping, which transforms the response output of artificial neurons and adds chaotic characteristics to its response output. Unlike ChaosNet [13], this paper does not directly use the emissivity of chaotic discharge characteristics of biological neurons, but makes the emissivity curve quasi-incremental. It is then used as the response output of chaotic neurons. The modeling process is described in detail in section 3.1.

3.1. Chaotic Neuron Modeling

Different from the traditional artificial neuron model, the chaotic neuron proposed in this paper is chaotic, random, dynamic, and time-dependent. This paper uses a one-dimensional chaotic map for its chaotic characteristics (see Background 2.3).

$q_t$represents the activity of chaotic neurons at discrete time t (activity can represent the membrane potential of biological neurons), which affects the activity of neurons at the next time t+1, which reflects the time dependence and dynamics of chaotic neurons of biological neurons, and $f(.)$represents a one-dimensional chaotic mapping function. The activity of the resting state of the current neuron $q_0$is not absolutely unchanged, and will be affected by the internal state of the neuron, showing a certain randomness. Assume a random value between 0 and 1: $q_0 = random(0,1)$.

$$q_{t+1} = f(q_t), \tag{3}$$

In the traditional artificial neuron model, the input neuron transmits the stimulus signal itself to the receiving neuron through the accumulation of synaptic weights. In chaotic neurons, what is transmitted to the receiving neuron is the emissivity (frequency of action potential generation) in the chaotic discharge characteristics, in which the emissivity is affected by the effective stimulation signal of the chaotic discharge of the excited neuron. Therefore, the emissivity of the input neuron to the receiving neuron is defined in this paper as:

$$f_t = \frac{1}{t}\sum_{i=1}^{t} 1_{\theta_t}(q_i) \tag{4}$$

$$1_{\theta_{t-1}}(q_i) = \begin{cases} 1, q_i \geq \theta_{t-1} \\ 0, q_i < \theta_{t-1} \end{cases} \tag{5}$$

θ_t represents the discharge threshold (the threshold for stimulating chaotic discharge). When it is assumed that chaotic neurons receive K stimulation signals from other

chaotic neurons, then the total stimulus received by chaotic neurons is:

$$h_t = \sum_{k=1}^{K} f_t^{(k)} \tag{6}$$

The total stimuli received by chaotic neurons affect their firing threshold θ_t

$$\theta_t = g(h_{t-1}). \tag{7}$$

The discharge threshold θ_t changes dynamically, at that time, θ_0 is the threshold value in the resting state of chaotic neurons. In biological neurons, the variability of the discharge threshold is not completely random, but has a certain regularity. In the lateral hypothalamus nucleus cells, 89% of the variability of the discharge threshold can be explained by the deterministic threshold formula [31]. Therefore, here we assume $g(\cdot)$a single decreasing function, which causes the adaptive phenomenon of threshold decline in biological neurons to appear, which will increase the frequency of chaotic neurons firing again, and thus its emissivity will gradually increase. When the total stimulus of chaotic neurons is higher, the lower the threshold, the more sensitive the chaotic neuron to the stimulus, which increases the sensitivity of the chaotic neuron to the stimulus. In addition, in order to make chaotic neurons have the saturation of biological neuron response, the threshold function $g(\cdot)$ no longer decreases continuously after the stimulus is strong enough, but tends to a stable value.

In general, formulas (3) to (7) represent a dynamic system with feedback: the intensity of the input stimulus affects the firing threshold of the chaotic neuron, which in turn affects the total intensity of the stimulus received by the chaotic neuron. When such dynamic systems are implemented as artificial neurons, they are complicated, and the computational efficiency of calculating the response of chaotic neurons is very low because of the need for repeated chaotic iterations. Therefore, the firing threshold of chaotic neurons is simplified here into a form that is only affected by the total input of chaotic neurons and does not change with time:

$$\theta = g(x) \tag{8}$$

x is the total stimulus input of chaotic neurons.

After such simplification, the chaotic response of chaotic neurons is reduced to a simple feedforward system: according to formula (8), the firing threshold of chaotic neurons is calculated according to the total input of the stimuli they receive; Chaotic neurons constantly update their own dynamic behavior state according to formula (3), and then calculate their emissivity according to formula (4) and (6), and serve as the final response output of chaotic neurons. Therefore, the chaotic neuron's response to input and output process can be simply expressed as an activation function:

$$y = chaotic_activation(x) \tag{9}$$

This activation function shows an overall growth trend with nonlinear quasi-increase and eventually tends to be stable. The image of this activation function can be viewed by 6. From the biotheological point of view, the output of chaotic neuron activation response has the characteristics of excitatory neuron response. The whole chaotic neuron modeling process has the complex dynamic behavior of biological neurons and the characteristics of chaotic system, and highly simulates the action potential generation process of biological neurons, and has biological interpretability.

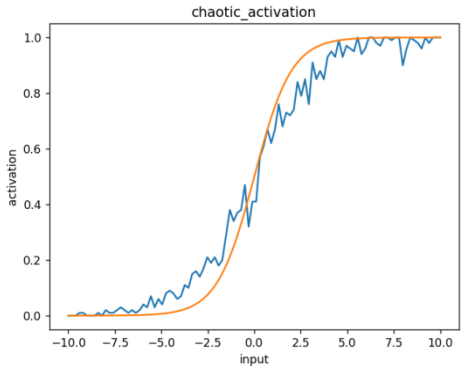


Figure 6. Image of chaotic neuron activation function curve

3.2. Properties of Chaotic Neurons and Their Activation Functions

For the selection of the threshold function $g(x)$, formula (10) is adopted in this paper according to the required characteristics to reflect the dynamic change process of the threshold with the increase of the total stimulus input of chaotic neurons:

$$g(x) = \frac{1}{1+e^{x}} \tag{10}$$

The selection of threshold function $g(x)$ satisfies the saturation behavior of chaotic neurons in response output. From the perspective of dynamics, the activation function curve of chaotic neurons in Figure 6 has the characteristics of biological neurons transforming external stimuli into peaks, and the whole is quasi-incremental. Since the activation function is generally differentiable, it is obvious that the activation function of chaotic neurons cannot be directly differentiated or differentiable. However, the activation function of chaotic neurons has saturation behavior, and the overall gradient direction is generally consistent with the sigmoid curve, while locally, the two are not completely consistent, which can avoid the network optimization falling into local optimality. This can help prevent overfitting to some extent. Therefore, we choose the derivative of sigmoid as the approximate derivative of chaotic neuron activation function, so equation (11) can be approximated as the derivative of chaotic neuron activation function.

$$f'(x) = \frac{1}{1+e^{-x}}\left(1-\frac{1}{1+e^{-x}}\right) \tag{11}$$

By modeling chaotic neurons and using activation functions to reflect their response outputs, we conclude that chaotic neurons and their activation functions have the following properties:

(1) Time dependence: The activity value of chaotic neurons depends on the activity value of the previous moment, which is similar to the time dependence of biological neurons on the input and output of signals, but artificial neurons do not;

(2) Response saturation: even if the input stimulus signal in chaotic neurons becomes larger and larger, its response no longer changes significantly, but maintains a relatively stable state, similar to the saturation behavior of biological neurons;

(3) Chaos and randomness: Due to the reference of chaos

mapping, the activity of chaotic neurons is random and more complex than that of artificial neurons;

(4) Dynamic adaptability: The internal threshold and activity of chaotic neurons change dynamically, and the threshold adaptively changes in response to the increase of total stimulus input, which is in line with the internal activity mechanism of biological neurons;

(5) Strong nonlinearity: the chaotic neuron activation function is more complex than the mainstream activation function and cannot be expressed by specific formulas. The chaotic neuron activation function has many inflection points, so it has strong nonlinearity, which can help the model better fit the data and improve the generalization ability of the model to a certain extent;

(6) Sensitive dependence: Because chaotic mapping is very sensitive to small changes in initial conditions, chaotic neurons will change when there is a small disturbance to the input. Therefore, the activation function of chaotic neurons is not fixed and will change significantly with the initial conditions every time.

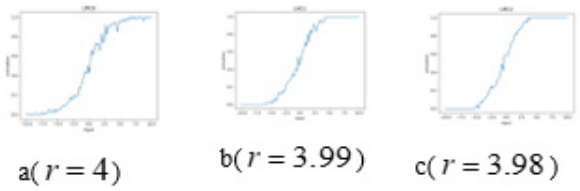


Figure 7. The peak number of chaotic neuron activation function curve changes obviously under different chaos control parameters

4. Experiment and Analysis

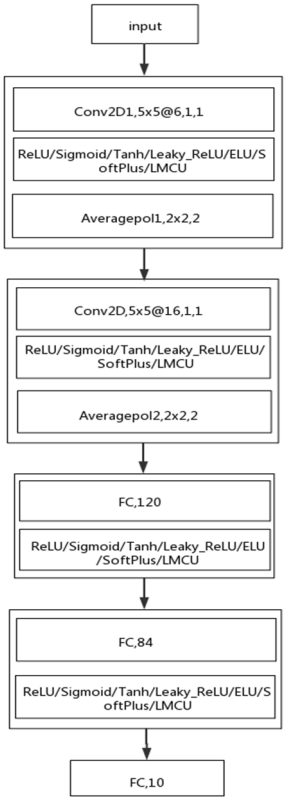


Figure 8. LeNet-5 model frame diagram

The experiment in this paper adopts tensorflow2 framework and python programming language, and is carried out on window10 operating system. Hardware platform: i7-10750H CPU, Nvidia Geforce GTX 1650 GP, 64GB memory. In this experiment, only Logistic Map in one-dimensional chaotic mapping was selected as the chaotic neuron model, and the chaotic neuron activation function proposed in this paper was named after LMCU (logistic map chaotic unit). In order to fully evaluate the learning performance and anti-overfitting ability of LMCU, experiments based on LeNet-5[34] deep learning model were conducted on the MNIST [32] and CIFAR-10[33] datasets, respectively. LMCU and sigmoid, ReLu, tanh, elu, Leaky_Relu and Softplus6 activation functions are compared and analyzed, among which the effects of different activation functions on LeNet-5 model can be seen in Figure 8. Finally, the experiment in this section will analyze the anti-overfitting ability of LMCU in the CIFAR-10 data set separately and compare it with the mainstream anti-overfitting means Dropout [35].

4.1. Results and Analysis of MNIST Dataset

The MNIST dataset consists of 48,000 training images, 12,000 validation images, and 10,000 test images, each of which is a 28×28 grayscale image. The data is preprocessed at the beginning of model training, and the training set and test set are directly divided by MNIST. In order to reflect the anti-overfitting ability of LMCU, this experiment started with the factors leading to the overfitting of the model, and conducted 100 rounds of overtraining on LeNet-5. In order to effectively analyze the performance of different activation functions in LeNet-5 model, the same hyperparameters were used for training. The hyperparameter design of the LeNet-5 model in this summary experiment was designed according to Table 1.

Table 1. MNIST-LeNet-5 model hyperparameter design

Model parameter design	
learning_data	0.001
batch_size	128
epochs	100
Cost function	Sparse classification cross entropy
optimizer	Adam

Under the hyperparameter design in Table 1, the last round of test set accuracy of each activation function was recorded after LeNet-5 model training. The learning performance of each activation function can be obtained by statistics in Table 2. It can be seen from Table 2 that the learning performance of the LMCU proposed in the paper in the LeNet-5 model of the MNIST dataset is basically comparable to that of mainstream activation functions.

Table 2. MNIST-LeNet-5 Comparison of learning performance of each activation function

Activation function	Accuracy %
relu	99.21
sigmoid	98.87
tanh	98.94
Leaky_relu	99.19
elu	99.01
softplus	98.83
LMCU	98.85

However, due to the chaotic cycle process in the output process of LMCU, the calculation time is longer and the training time is slower than those of the main activation

functions under the same number of training rounds.

In Table 3, the difference of accuracy and loss rate of different activation functions in the training set and the test set of LeNet-5 model is recorded. Slight overfitting occurs in the overtrained LeNet-5 (the performance of the test set is worse than that of the training set as a whole), even though the gradient direction of the LMCU curve is consistent with sigmoid. However, it can still be found that LMCU has the lowest generalization error and the lowest degree of overfitting, which indicates that LMCU has the best anti-overfitting ability compared with other mainstream activation functions.

Table 3. MNIST-LeNet-5 Comparison of generalization errors of each activation function

Activation function	Accuracy error (-)	Loss rate error (+)
relu	0.0079	0.0575
sigmoid	0.0113	0.0523
tanh	0.0106	0.0551
Leaky_relu	0.0081	0.0575
elu	0.0088	0.0796
softplus	0.0108	0.1047
LMCU	0.0071	0.0235

4.2. Results and Analysis of CIFAR-10 Dataset

The CIFAR-10 dataset is made up of 60,000 RGB images with a size of 32×32, and contains 10 different attributes of species such as birds, cars, and airplanes. Compared to the MNIST dataset, although CIFAR-10 is also 10 categories, it is more difficult to identify because its images contain more additional information. The data is preprocessed at the beginning of model training, and the training set and test set are also directly divided using the CIFAR-10 data set. LeNet-5 was trained for 50 rounds. In order to objectively compare the performance of LMCU with other mainstream activation functions, LeNet-5 model hyperparameter design in Table 4 was used for model training.

Table 4. Hyperparameter design of CIFAR-10-LENET-5 model

Model parameter design	
learning_data	0.001
batch_size	128
epochs	50
Cost function	Sparse classification cross entropy
optimizer	Adam

Under the hyperparameter design in Table 4, the last round of test set accuracy of each activation function was recorded after LeNet-5 model training. The learning performance of each activation function can be obtained by statistics in Table 5. It can be seen from Table 5 that relu performs best in LeNet-5, while LMCU performs worst in the response curve of chaotic neurons, with an accuracy 9.21% lower than relu and 5.77% lower than sigmoid. The accuracy rate of LMCU is close to that of softplus. Even though LMCU, sigmoid and tanh all have saturation behavior, it indicates that different activation functions have different performances in different data sets under the same model. Therefore, different activation functions should be selected under different data sets to improve the expression ability of the model. Because the data set CIFAR-10 is more complex than MNIST, it is a three-channel RGB image, and the image is more susceptible to lighting, object pose, occlusion and other factors. Therefore, the classification performance of LeNet-5 model will also decline.

Table 5. CIFAR-10-LeNet-5 Learning performance of each activation function

Activation function	Accuracy %
relu	60.49
sigmoid	57.05
tanh	53.34
Leaky_relu	60.32
elu	56.57
softplus	51.40
LMCU	51.28

In Table 6, by comparing the difference of accuracy and loss rate of different activation functions in the training set and test set of LeNet-5 model, overfitting occurred in LeNet-5 after 50 rounds of training. It is obvious that LMCU has the lowest generalization error, with the accuracy error and loss rate error of only 1.1% and 3.06% respectively in the training set and test set. The degree of overfitting was the lowest. This shows that the LeNet-5 model based on LMCU activation function has the best resistance to overfitting and the strongest generalization ability compared with the LeNet-5 model based on other mainstream activation functions.

Table 6. CIFAR-10-LeNet-5 Comparison of generalization errors of each activation function

Activation function	Accuracy error (-)	Loss rate error (+)
relu	0.2258	1.0495
sigmoid	0.0449	0.119
tanh	0.2735	1.1625
Leaky_relu	0.2046	0.8678
elu	0.2971	1.7323
softplus	0.3161	2.169
LMCU	0.011	0.0306

4.2.1. Analysis of Anti-overfitting Ability

In order to further analyze the anti-overfitting ability of LMCUs, this section continues to conduct comparative experiments on the influence of different chaos parameters on LeNet-5 model based on relu activation function in CIFAR-10 data set. First, the LeNet-5 model based on relu activation function was trained according to the hyperparameters in Table 4. The curves of accuracy rate and loss rate of the training set and test set were printed respectively, 9. It can be seen that the model had serious overfitting, and the accuracy rate of the last round of test set was 0.6268.

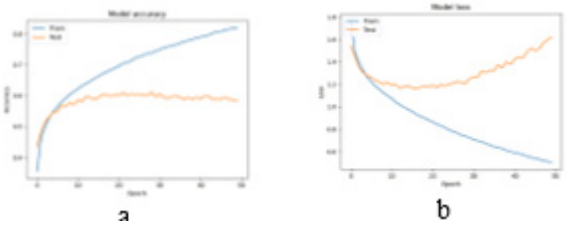


Figure 9. relu-LeNet-5 Performance of training set and test set

Then, in the two-layer convolution of LeNet-5 based on relu activation function, the relu activation function is successively replaced with chaos control parameters with step size of 0.1: The LMCUs of 4, 3.9 and 3.8 were then trained respectively, and the accuracy rate of the last round of test set was recorded respectively at 0.6020, 0.5779 and 0.5785. The accuracy rate and loss rate errors of the training set and test

set of different chaos control parameters were recorded in Table 7. At the same time, the curves of precision rate and loss rate of the training set and test set with chaos control parameter 4 are printed, and compared with Figure 9, it can be found that the overfitting degree is effectively suppressed by the replacement of LMCU.

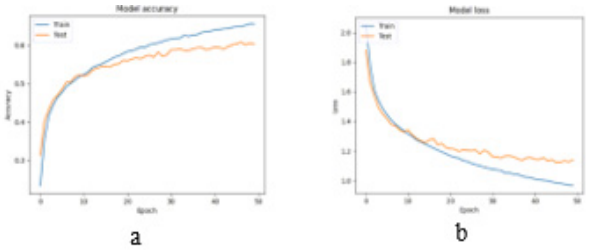


Figure 10. LMCU training set and test set performance with chaos control parameter 4

In Table 7, the replacement performance of LMCU's two-layer convolution in LeNet-5 when the chaos control parameters are 4, 3.9, and 3.8 are respectively recorded. It can be concluded that as the chaos control parameters become smaller, the lower the degree of chaos, the larger the generalization error, and the lower the ability to prevent overfitting. Therefore, the anti-overfitting ability of LMCU is affected by the control chaos control parameter. The larger the chaos control parameter, the stronger the anti-overfitting ability of LMCU. As can be seen from Table 7, when the chaos control parameter is 4, LMCU has the best anti-overfitting ability and the lowest generalization error. In Figure 7, it is found that chaos control parameters affect the peak number of LMCU. The larger the chaos parameter, the more the peak number of LMCU, which also indicates that the more the peak number of LMCU, the stronger the anti-overfitting ability of LMCU.

Table 7. Comparison of generalization errors under different chaos control parameters

Chaos control parameter (r)	Accuracy error (-)	Loss rate error (+)	The final test set accuracy
4	0.0236	0.0903	0.6020
3.9	0.0572	0.1559	0.5779
3.8	0.086	0.2755	0.5785

4.2.2. Comparison of Anti-overfitting Ability

Dropout is a common anti-overfitting method. Dropout randomly drops the output of certain neurons with a certain probability during training, and each layer of the neural network will no longer be overly dependent on a particular neuron, which reduces the risk of overfitting. Dropout is often applied after the convolution layer and the fully connected layer. In the LeNet-5 model based on relu activation function, experiments in this section randomly discard neurons in the convolution layer and the fully connected layer of LeNet-5 with probability values P of 0.2, 0.5 and 0.6, respectively, and then carry out the same training. The generalization error of LeNet-5 after adding Dropout and the accuracy of the last test set were also recorded.

Table 8 shows that when the Dropout probability value p is 0.2, LeNet-5 has the lowest generalization error, the lowest overfitting degree, and the highest accuracy rate. Compared with the use of LMCU when the chaos control parameter is 4 in Table 7, the anti-overfitting ability of Dropout is slightly superior. However, the use of Dropout involves the adjustment of different probability values and the adjustment of different positions, which greatly reduces the efficiency of model training. Compared with adding some anti-overfitting methods to the model, the application of chaotic neuron response curve LMCU in the artificial neural network can reduce the dependence of the artificial neural network on the mainstream anti-overfitting methods, so as to improve the efficiency of model training. It starts from the artificial neural network itself, so as to enhance the anti-overfitting ability of the artificial neural network.

Table 8. Performance of different probability values P

Dropout(P)	Accuracy error (-)	Loss rate error (+)	The final test set accuracy
0.2	0.0019	0.0055	0.6204
0.5	0.0283	0.0934	0.5182
0.6	0.0423	0.1585	0.4668

5. Conclusion

Inspired by the chaotic discharge characteristics of biological neurons and based on the quasi-incremental characteristics of chaotic and nonlinear responses of biological neurons, this paper proposes a novel chaotic neuron with chaotic response characteristics and its activation function LMCU. This chaotic neuron is based on Logistic Map. At the same time, the complex dynamic behavior of biological neurons and the characteristics of chaotic system are combined, and the emissivity of chaotic firing characteristics of biological neurons is taken as the response output of chaotic neurons. The LMCU activation function curve has the characteristics of numerous spikes that the traditional activation function curve does not have, and its gradient is roughly consistent with sigmoid in the general direction. Comparative experiments using LeNet-5 model on data sets MNIST and CIFAR-10 draw the following conclusions:

(1) The learning ability of LMCU is basically comparable to the mainstream activation function.

(2) Compared with the mainstream activation function, the use of LMCU can effectively suppress the overfitting of artificial neural network, greatly reduce the generalization error of the model, and enhance the anti-overfitting ability of artificial neural network.

(3) The anti-overfitting ability of LMCU is affected by the chaos control parameters. When the chaos control parameters are larger, the system is more chaotic, the LMCU curve has more spikes, and the anti-overfitting ability of LMCU is better.

(4) In the comparison experiment with Dropout, the anti-overfitting ability of Dropout is slightly superior, but the use of Dropout involves the adjustment of different probability values and the adjustment of different positions, which greatly reduces the efficiency of model training. Compared with adding some anti-overfitting means to the model, the application of LMCU in artificial neural network can reduce the dependence of artificial neural network on mainstream anti-overfitting means. It starts from the artificial neural network itself, so as to enhance the anti-overfitting ability of artificial neural network.

(5) The response output process of chaotic neurons involves the calculation of chaotic iterative cycles, so its response curve LMCU has a problem of slow computation efficiency in model training.

In this paper, only Logistic Map was used for the modeling and experimental study of chaotic neurons, and Tent Map was not used for experimental study. In the next research, we will use Tent Map to realize the chaotic neuron modeling and its activation function. At the same time, there are still many places to explore for this new chaotic neuron model proposed in this paper.

Acknowledgments

This work is supported by the Sichuan Science and Techn ology Program (No. 23ZDYF0380, No. 2021ZYD0011).

References

[1] Tan, Mingxing, and Quoc V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. International Conference on Machine Learning (ICML), 2020, pp. 6105-6114.

[2] Dosovitskiy, A.,Beyer, L.,Kolesnikov,A.,et al.An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.ArXiv preprint arXiv:2010.11929 (2020).

[3] HODGKIN AL, HUXLEY AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. J Physiol. 1952 Aug;117(4):500-44. doi: 10.1113/jphysiol.1952.sp004764. PMID: 12991237; PMCID: PMC1392413.

[4] Mai Ying-chao, Chen Yun-Hua, Zhang Ling. Strong anti-noise neuronal activation function with biological authenticity [J]. Computer science, 2019, 46 (7) : 206-210. https:// doi. org/ 10. 11896/j.issn.1002-137X.2019.07.031

[5] Kathleen T Alligood, Tim D Sauer, and James A Yorke. Chaos. Springer, 1996.

[6] Walter J Freeman et al. Mass action in the nervous system, volume 2004. Citeseer, 1975.

[7] Hung-Jen Chang and Walter J Freeman. Parameter optimization in models of the olfactory neural system. Neural Networks, 9(1):1–14, 1996.

[8] Robert Kozma and Walter J Freeman. A possible mechanism for intermittent oscillations in the kiii model of dynamic memories-the case study of olfaction. In IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No. 99CH36339), volume 1, pages 52–57. IEEE, 1999.

[9] Ichiro Tsuda. Dynamic link of memory—chaotic memory map in nonequilibrium neural networks. Neural networks, 5(2): 313-326, 1992.

[10] John S Nicolis and Ichiro Tsuda. Chaotic dynamics of information processing: The "magic number seven plus-minus two" revisited. Bulletin of Mathematical Biology, 47(3):343–365, 1985.

[11] Kunihiko Kaneko. Lyapunov analysis and information flflow in coupled map lattices. Physica D: Nonlinear Phenomena, 23 (1-3):436–447, 1986.

[12] Kunihiko Kaneko. Clustering, coding, switching, hierarchical ordering, and control in a network of chaotic elements. Physica D: Nonlinear Phenomena, 41(2):137–172, 1990.

[13] Harikrishnan Nellippallil Balakrishnan, Aditi Kathpalia, Snehanshu Saha, and Nithin Nagaraj. Chaosnet: A chaos based artifificial neural network architecture for classifification. Chaos: An Interdisciplinary Journal of Nonlinear Science, 29(11):113125, 2019.

[14] N.B. Harikrishnan,Nithin Nagaraj, Neurochaos inspired hybrid machine learning architecture for classification, in: 2020 International Conference on Signal Processing and Communications, SPCOM, IEEE, 2020, pp. 1–5.

[15] Sethi D, Nagaraj N, Harikrishnan N. Neurochaos feature transformation for machine learning. Integration 2023.

[16] Remya Ajai A.S.,Harikrishnan N.B. , Nithin Nagaraj .Analysis of logistic map based neurons in neurochaos learning architectures for data classification 2023.

[17] Nair, Vinod, and Geoffrey E. Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines.Proceedings of the 27th International Conference on Machine Learning (ICML), 2010, pp. 807-814.

[18] Maas, Andrew L., Awni Y. Hannun, and Andrew Y. Ng. Rectifier Nonlinearities Improve Neural Network Acoustic Models. Proceedings of the 30th International Conference on Machine Learning (ICML), vol. 30, no. 1, 2013.

[19] Clevert, Djork-Arné, Thomas Unterthiner, and Sepp Hochreiter. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). International Conference on Learning Representations (ICLR), 2015.

[20] Ramachandran, Prajit, Barret Zoph, and Quoc V. Le. Swish: A Self-Gated Activation Function.Proceedings of the 35th International Conference on Machine Learning (ICML), 2017.

[21] Goodfellow, Ian J., David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout Networks.Proceedings of the 30th International Conference on Machine Learning (ICML), 2013, pp. 1319-1327.

[22] Misra D. Mish:a self regularized non-monotonic neural activation function [J].Ar Xiv,2020:1908.08681v3.

[23] Liu Yuqing, Wang Tian-Hao, Xu Xu. Novel adaptive activation function for deep learning neural networks [J]. Journal of Jilin University (Science Edition), 2019, 57(04): 857-859.

[24] Xu Jingping, Wang Fang. Image classification method based on improved S-ReLU activation function [J]. Science Technology and Engineering,2022,22(29):12963-12968.

[25] Zhang Huan, Zhang Qing, Yu Jiyan. Property Analysis and improvement of activation Function in Convolutional Neural Networks [J]. Computer Simulation,202,39(4):328-334.

[26] Matthew Mithra Noel, Shubham Bharadwaj, Venkataraman Muthiah-Nakarajan, Praneet Dutta, and Geraldine Bessie Amali. Biologically inspired oscillating activation functions can bridge the performance gap between biological and artifificial neurons. arXiv preprint arXiv:2111.04020, 2021.

[27] Lu Q, Tian J. Synchronization and stochastic resonance of the small-world neural network based on the CPG. Cogn Neurodyn. 2014 Jun;8(3):217-26. doi: 10.1007/s11571-013-9275-8. Epub 2013 Nov 13. PMID: 24808930; PMCID: PMC4012065.

[28] Muscinelli, SP, Gerstner, W, Schwalger, T. How Single Neuron Properties Shape Chaotic Dynamics and Signal Transmission in Random Neural Networks. PLOS Computational Biology, vol. 15, no. 6, 2019, e1007122. https:// doi.org/ 10.1371/journal.pcbi.1007122.

[29] Valle, João, et al. Chaotical PRNG based on composition of logistic and tent maps using deep-zoom. Chaos, Solitons& Fractals, vol. 161,2022,112296.ISSN0960-0779. doi: 10. 1016/ j. chaos. 2022.112296.

[30] LUO Yuling, LI Tian-hao, XIAO Dingwei, QIU Sen-hui. A Chaotic cryptosystem resistant to energy Analysis attacks [J]. Journal of Hunan University: Natural Science Edition, 2012, 49 (4):47~57

[31] Yi Guosheng, Zhao Qiang, Wei Xile, et al. Variability of neuronal firing threshold and its significance [J]. Progress in Biochemistry and Biophysics, 2019,49(07):1226-1242.

[32] LeCun Y, Léon Bottou L, Bengio Y, Haffner P. Gradient-Based Learning Applied to Document Recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[33] Krizhevsky A,Hinton G.Learning multiple layers of features from tiny images[J].Handbook of Systemic Autoimmune Diseases, 2009,1(4):1-60.

[34] LeCun, Yann, et al. Gradient-Based Learning Applied to Document Recognition.Proceedings of the IEEE, vol. 86, no. 11, 1998, pp. 2278-2324.

[35] Srivastava, Nitish, et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research, vol. 15, no. 1, 2014, pp. 1929-1958.