Proposal for new type function in deep neural network

Hwakyeom Kim

hwakyeom@icloud.com

Abstract

I'll represent a new-type activation unit named TIUD(Tuning Input Unit from Distribution) using the distribution of input data. This tunes each data in the batches. Their $x^{(i)}$ having standard deviation and mean. $\mu_{X_i} * \sigma_{X_i}$ is denoted 'Id'. So deep neural network can choose how much do active datas. Also had tested amount of samples, get the results of succeed.

Introduction

The neural net area has a couple of activation functions like ReLU, LeakyReLU, PReLU etc. Lately GELU seems to be quietly useful. I wonder that is there optimal function on this area? People often say "Well, A.I will destroy us using their Auto-algorithm something whatever." Yes, but i hope no.

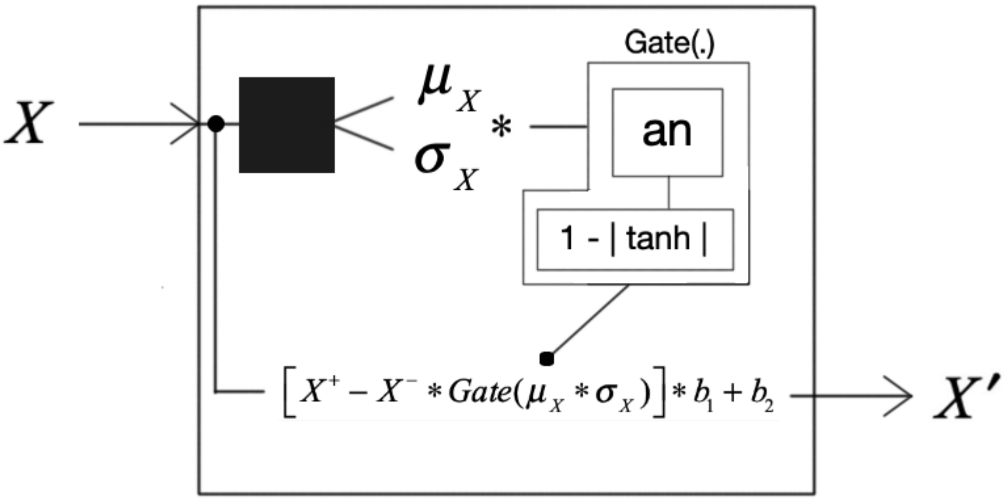
A novel type function

$$X^{(i)} = \begin{pmatrix} x_{11} & x_{12} & \cdots \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & x_{mn} \end{pmatrix} \qquad [X] = \begin{pmatrix} [x_{11} & \cdots & x_{mn}]^{(1)} \\ [x_{11} & \cdots & x_{mn}]^{(2)} \\ [x_{11} & \cdots & x_{mn}]^{(b)} \end{pmatrix}$$

Here is $X^{(i)}$ information and their batches set. $[X] \rightarrow TIUD \rightarrow [X']$, i'll explain. First of all μ and σ (according to previous denoting, each mean mean and standard deviation) are based on mini-batches. It means we apply Batch normalization before extract each data's μ_{X_i} and σ_{X_i}. Batch normalizing formulates data's identity.

Multiplying them we will call ***values***.

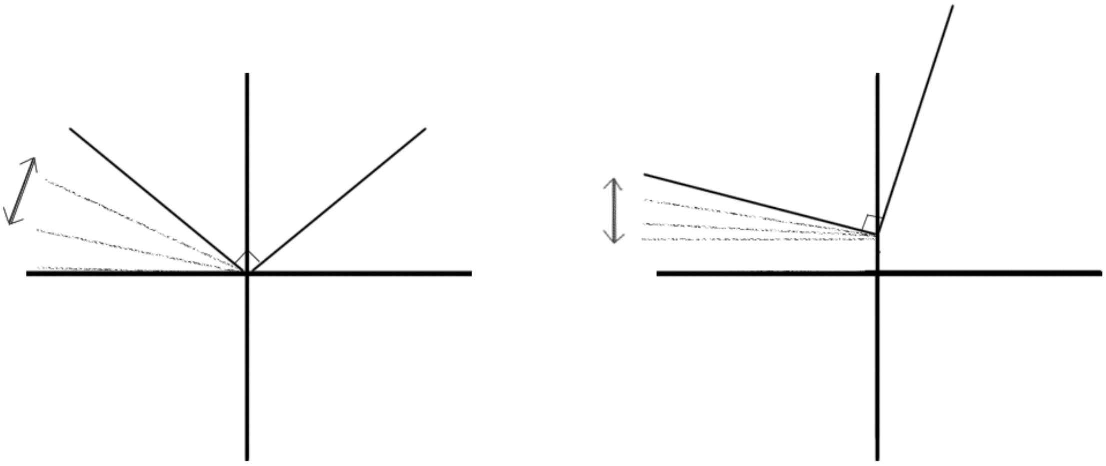
And this is the architecture of TIUD.


X
μ_X
σ_X
*
Gate(.)
an
1 - | tanh |
$\left[X^{+} - X^{-} * Gate(\mu_X * \sigma_X)\right] * b_1 + b_2$
X'

$$TIUD(x) = \left[X^{+} - X^{-} * Gate(\mu_X * \sigma_X)\right] * b_1 + b$$

* : elementwise product

And this is a simple graph of activation.



X^{+} : as the positive part of its input

X^{-} : as the negative part of its input

$Gate(\cdot)$: Zero to One

b_1, b_2 : final biases for deep neural net

Main thing is Gate(˙).

Gate(˙) has two parts, describes **an**(˙) and **ac**(˙). - **an**alysis and **ac**tive

$$an(Id) = W_{\beta} \times \left(W_{\alpha} \times Id + b_{\alpha}\right) + b_{\beta}$$

$$ac(an) = 1 - |\tanh(an)|$$

Finally we get this formula.

$$T(x) = b_1 * \left[X^{+} - X^{-} * \left(1 - |\tanh(W_{\beta} \times W_{\alpha} \times \mu_X \sigma_X + W_{\beta} \times b_{\alpha} + b_{\beta})|\right)\right] + b_2$$

※ b_1, b_2 also important things. These make the network to learn easily. Where units are, in terms of, $\textit{learning rate}^{(i)}$ and $\textit{data size}^{(i)}$ change automatically.