Chapter 08. 좋은 딥러닝 구조를 찾아내는 딥러닝 (Neural Architecture Search)

Activation Functions

Searching For Activation Functions

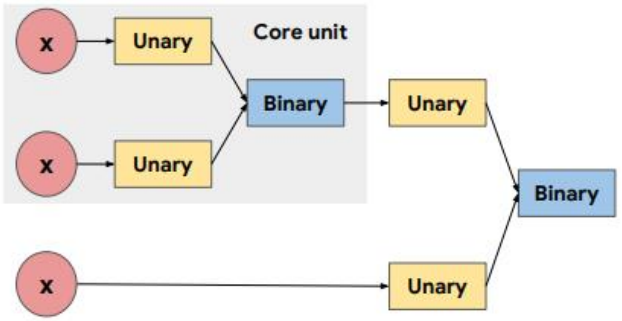
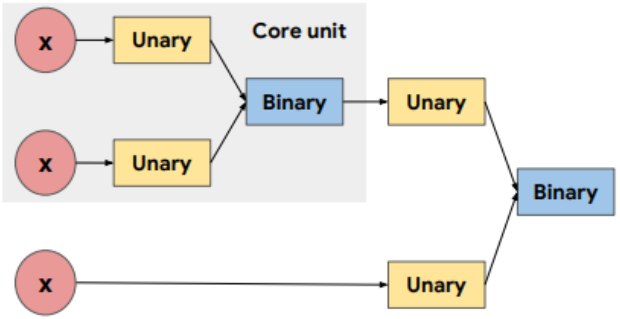


Figure 1: An example activation function structure. The activation function is composed of multiple repetitions of the "core unit", which consists of two inputs, two unary functions, and one binary function. Unary functions take in a single scalar input and return a single scalar output, such $u(x) = x^2$ or $u(x) = \sigma(x)$. Binary functions take in two scalar inputs and return a single scalar output, such as $b(x_1, x_2) = x_1 \cdot x_2$ or $b(x_1, x_2) = \exp(-(x_1 - x_2)^2)$.

ReLU를 뛰어넘는 Activation Function을 찾기 위한 흥미로운 연구로, Google Brain 연구다.

Search Space



- **Unary functions**: x, $-x$, $|x|$, x^2, x^3, $\sqrt{x}$, βx, $x+\beta$, $\log(|x|+\epsilon)$, $\exp(x)$ $\sin(x)$, $\cos(x)$, $\sinh(x)$, $\cosh(x)$, $\tanh(x)$, $\sinh^{-1}(x)$, $\tan^{-1}(x)$, $\mathbf{sinc}(x)$, $\max(x,0)$, $\min(x,0)$, $\sigma(x)$, $\log(1+\exp(x))$, $\exp(-x^2)$, $\mathbf{erf}(x)$, β
- **Binary functions**: x_1+x_2, $x_1 \cdot x_2$, x_1-x_2, $\frac{x_1}{x_2+\epsilon}$, $\max(x_1,x_2)$, $\min(x_1,x_2)$, $\sigma(x_1)\cdot x_2$, $\exp(-\beta(x_1-x_2)^2)$, $\exp(-\beta|x_1-x_2|)$, $\beta x_1+(1-\beta)x_2$

Unary Function들과 Binary Function들을 미리 선정해 Search Space를 구성하였다.

RNN Controller

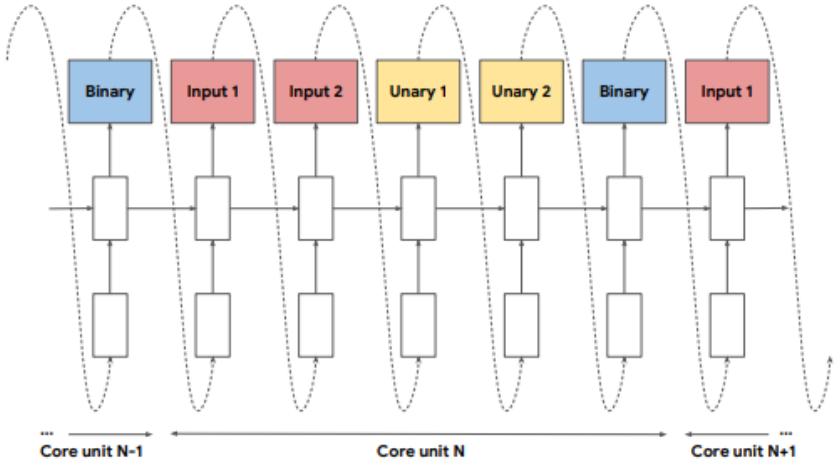


Figure 2: The RNN controller used to search over large spaces. At each step, it predicts a single component of the activation function. The prediction is fed back as input to the next timestep in an autoregressive fashion. The controller keeps predicting until every component of the activation function has been chosen. The controller is trained with reinforcement learning.

RNN Controller를 사용해 Reinforcement Learning을 수행하였다.

Top Novel Activations

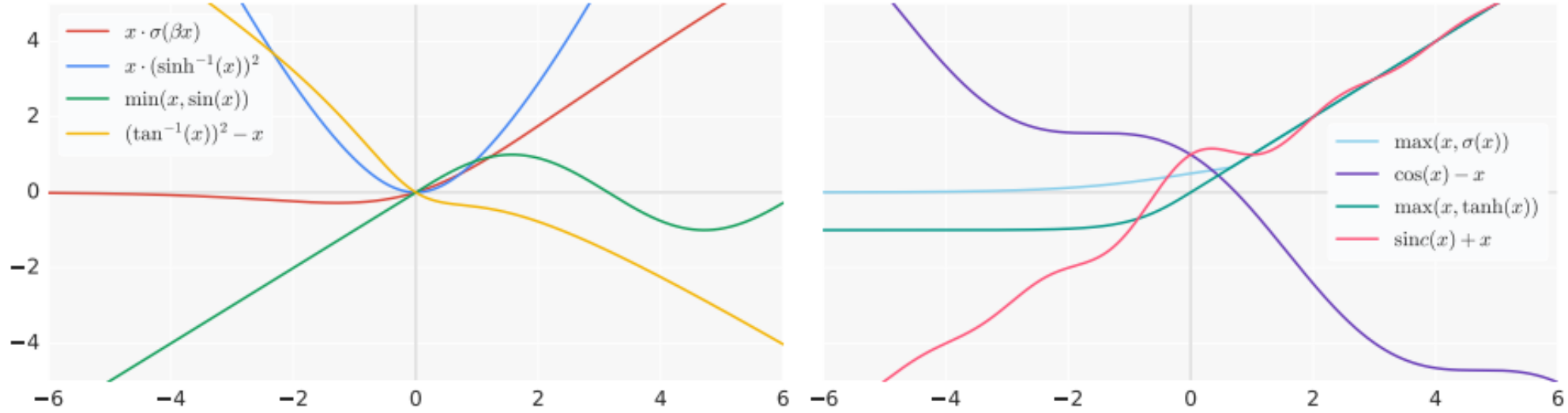


Figure 3: The top novel activation functions found by the searches. Separated into two diagrams for visual clarity. Best viewed in color.

이 중 최고 성능을 나타낸 함수는 $x\sigma(\beta x)$로, 해당 논문에서 Swish라고 명명하였다.

Results

Model	ResNet	WRN	DenseNet
LReLU	94.2	95.6	94.7
PReLU	94.1	95.1	94.5
Softplus	94.6	94.9	94.7
ELU	94.1	94.1	94.4
SELU	93.0	93.2	93.9
GELU	94.3	95.5	94.8
ReLU	93.8	95.3	94.8
Swish-1	94.7	95.5	94.8
Swish	94.5	95.5	94.8

Table 4: CIFAR-10 accuracy.

Model	ResNet	WRN	DenseNet
LReLU	74.2	78.0	83.3
PReLU	74.5	77.3	81.5
Softplus	76.0	78.4	83.7
ELU	75.0	76.0	80.6
SELU	73.2	74.3	80.8
GELU	74.7	78.0	83.8
ReLU	74.2	77.8	83.7
Swish-1	75.1	78.5	83.8
Swish	75.1	78.0	83.9

Table 5: CIFAR-100 accuracy.

무난하게 평균적으로 ReLU를 뛰어넘는 성능을 보여준다. TF2.1 기준 `tf.nn.swish`로 사용할 수 있다.