Check for updates

Journal of KIIT. Vol. 19, No. 1, pp. 1-9, Jan. 31, 2021. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2021.19.1.1

CoS: 딥러닝을 위한 시그모이드로 구성한 강조된 부드러운 비단조성 활성 함수

한준*, 강대성**

CoS: An Emphasized Smooth Non-Monotonic Activation Function Consisting of Sigmoid For Deep Learning

Jun Han*, Dae-Seong Kang**

이 논문은 2017 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.2017R1D1A1B04030870)

요 약

활성 함수는 신경망의 성능에 영향을 주는 중요한 구성요소이다. 기존의 계단, 시그모이드, 하이퍼볼릭 탄젠트, ReLU와 같은 활성 함수들은 학습에 중요한 미분이 되지 않거나 기울기가 소실되는 등의 문제점이 있다. 하지만 Swish 활성 함수는 비단조성을 가지는 부드러운 곡선과 음의 경계값을 가지는 특성으로 기존의 문제점을 해결하였다. 이와 같은 특성을 기반하여 본 논문에서는 부드러운 곡선 부분을 추가하여 비단조성을 더욱 강조하고 음의 경계값을 가지는 특성을 줄이면서 신경망 안에서 정보의 흐름을 개선한 CoS(Consisting of Sigmoid)함수를 제안한다. 제안하는 방법을 통해 기존의 ReLU 보다 0.46%~0.77%, Swish보다는 0.38%~0.54%의 정확도 개선에 따른 성능 향상을 확인하였다.

Abstract

Activation functions are important components that affect the performance of neural networks. Activation functions such as Step, Sigmoid, Tanh, and ReLU have problems such that the important differentiation is not possible or the gradient is vanishing. Activation function Swish solved the problem with Characteristic of a smooth non-monotonic curves and boundary value of negative. Based on these characteristics, in this paper, we propose a Consisting of Sigmoid(CoS) function adding a smooth emphasized non-monotonic curves part and reducing negative results on negative inputs and improving the flow of information within the neural network. Through the proposed method, it was certain that accuracy is improved by 0.46%~0.77% and 0.38%~0.54% over the existing ReLU and Swish.

Keywords
activation function, deep learning, CNN, neural network, object detection

* 동아대학교 전자공학과 석사과정
- ORCID: https://orcid.org/0000-0002-7287-858X
** 동아대학교 전자공학과 교수(교신저자)
- ORCID: https://orcid.org/0000-0003-0186-2430

· Received: Sep. 21, 2020, Revised: Jan. 14, 2021, Accepted: Jan. 17, 2021
· Corresponding Author: Dae-Seong Kang
Dept. of Dong-A University, 37 NaKdong-Daero 550, beon-gil saha-gu, Busan, Korea,
Tel.: +82-51-200-7710, Email: dskang@dau.ac.kr

Ⅰ. 서 론

AI 분야 중 딥러닝은 이미지 인식, 음성 인식, 문장 예측 등 수많은 분야에서 다양하게 개발 및 연구가 진행되고 있다. 본 논문에서는 이미지 인식 분야를 다뤄 보려고 한다. 현재 이미지에 대한 인식(Recognition) 및 분류(Classification) 분야에 가장 흔히 쓰이는 알고리즘은 이미지의 공간 정보를 유지한 상태로 학습이 가능한 CNN(Convolution Neural Network)이다. 이는 심층 신경망의 한 종류로 동물의 시각 피질에서의 생물학적 작동 과정을 통해 영향을 받아 구현되었다[1].

기본적인 구성으로 그림 1과 같이 여러 개의 합성곱 계층과 풀링 계층, 완전 연결 계층 등으로 구성되어있다. 이미지 특징 추출을 위하여 그림 1과 같이 입력 데이터를 필터가 순회하며 합성곱을 계산하고, 계산 결과를 이용하여 특징 맵을 만든다[2].

CNN을 기반으로 한 VGGNet[3]과 GoogLeNet[4] 등과 같은 다양한 네트워크는 합성곱 계층이 여러 층이 쌓여 있는 모델이다. 신경망이 깊게 구성되어 있어 적용되는 활성 함수의 수도 많다. 그러기 때문에 활성 함수는 신경망의 성능에 영향을 주는 중요한 구성요소라 할 수 있다. 또한, 딥러닝의 학습에서 입출력의 관계가 복잡해야 학습에 대한 성능이 향상되기 때문에 비선형 함수의 사용이 필수적이다. 초기 활성 함수로 시그모이드나 하이퍼볼릭 탄젠트를 주로 사용했지만 양 끝단에 수렴이 될 때 기울기 소실 현상으로 학습에 대한 어려움이 존재하였다. 그래서 등장한 ReLU는 계산이 간단하고, 학습도 빨라 비효율성을 해결해주면서 딥러닝 모델들의 성능이 상당히 발전하게 된다. 하지만 음수에서 미분값이 0이라 학습이 불가능해지는 Dying ReLU 현상이 있어 이런 단점을 개선하기 위해 다양한 활성 함수가 연구되었다. 그 중 Swish 함수가 부드러운 비단조성 곡선과 음의 경계값을 가지는 특성으로 대부분의 단점을 개선하였다.

본 논문에서는 Swish 함수의 기반에 추가적인 시그모이드 제곱식을 통해 매끄러운 곡선을 추가함으로써 비단조성과 비선형성을 더욱 확보하고 입출력 관계의 복잡성을 높여 새로운 활성 함수(CoS, Consist of Sigmoid)를 제안한다.

Ⅱ. 관련 이론

2.1 활성 함수

인공 신경망의 기본 구성은 입력층, 은닉층, 출력층으로 나누어져 있다. 특히, 활성 함수[5]는 심층 신경망에서 중요한 역할을 하는 것 중 하나이고 심층 신경망에서 뉴런들에 들어가는 입력 신호의 총합을 그대로 사용하지 않고, 출력 신호로 변환하는 함수이다.

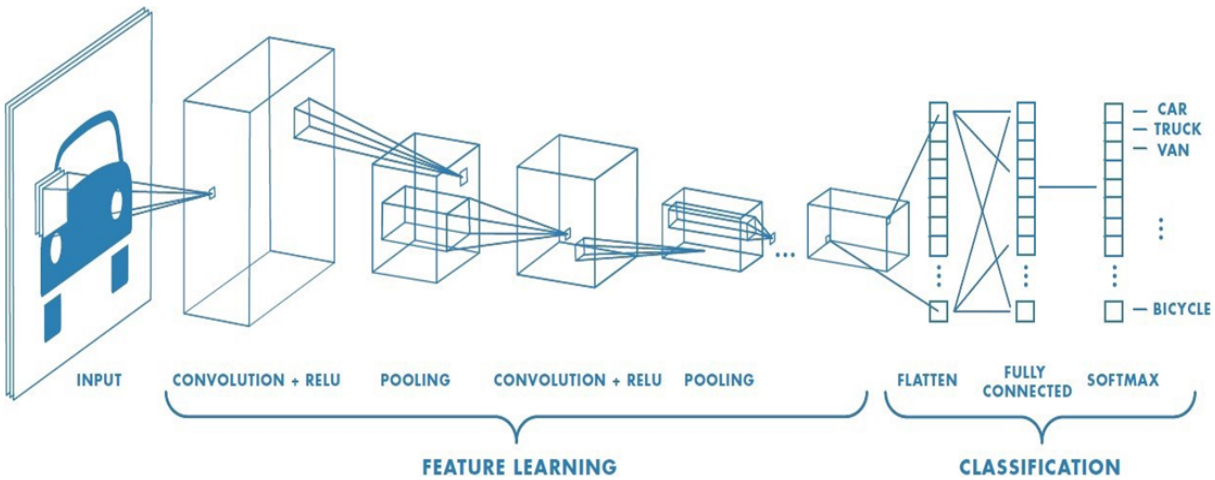


그림 1. CNN의 기본적인 구조
Fig. 1. Basic structure of CNN

신경망에서는 뉴런에 연산 값을 그림 2와 같이 계속 전달해주는 방식으로 가중치를 훈련한다. 각각의 함수는 네트워크의 각 뉴런에 연결되어 있으며, 각 뉴런의 입력이 모델의 예측과 관련되어 있는지에 따라 입력값에서 필요한 정보를 학습한다.

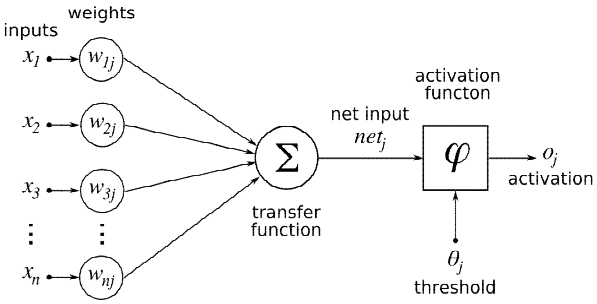


그림 2. 일반적인 신경망 Layer의 구조
Fig. 2. Structure of general neural network layer

또한, 심층 신경망의 학습 과정에서 순전파와 역전파를 그림 3과 같이 구성된다. 순전파 학습 과정은 입력층부터 은닉층을 통과하여 출력층까지 순방향으로 진행한다. 역전파 학습 과정은 순전파 학습 과정과 반대의 순서로 진행된다. 심층 신경망을 학습함으로써 활성 함수는 각 층에 밀접하게 관련이 되어있기 때문에 학습 속도나 성능 등 연산에 대한 효율성이 중요하다[6].

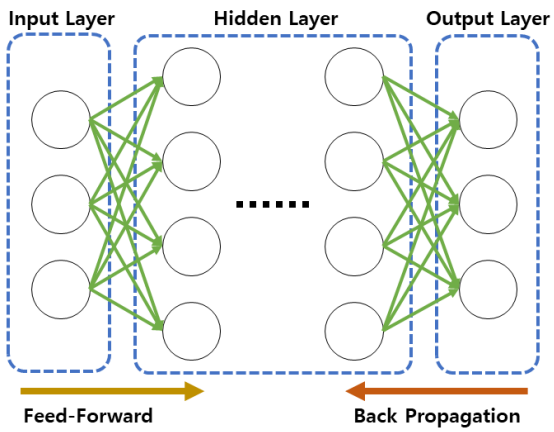


그림 3. 심층 신경망의 구조와 학습
Fig. 3. Structure and learning of deep neural networks

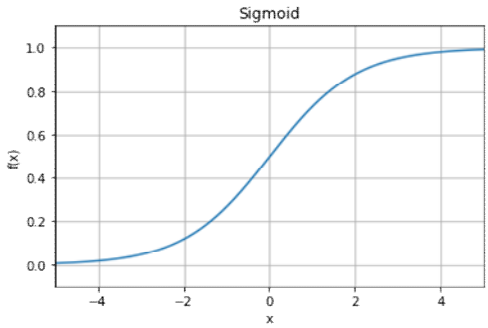
활성 함수의 종류로는 크게 두 가지로 선형 함수와 비선형 함수로 나눌 수 있다. 기본적으로 역전파를 학습할 때 활성 함수를 미분하여 이를 이용해 손실 값을 줄이는 과정을 거친다. 하지만 선형 함수의 미분 값은 상수이기에 입력값과 상관없는 결과를 얻는다. 그래서 예측과 가중치에 대한 상호관계에 대한 정보를 얻을 수 없기에 대부분 활성 함수로 비선형 함수를 사용하여 출력값을 다음 계층에 전달한다.

2.1.1 시그모이드 함수

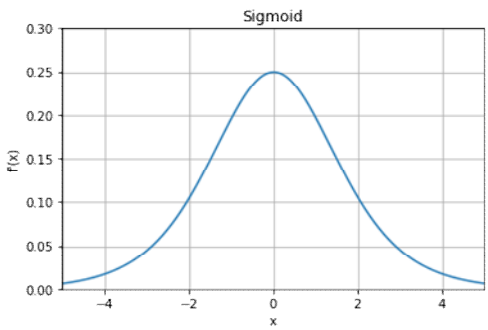
로지스틱으로 불리는 시그모이드 함수는 s자 형태를 띄는 함수이며 식 (1)과 같고 그림 4와 같은 그래프 형태를 볼 수 있다.

$$f(x) = \frac{1}{1+e^{-x}} = \sigma(x) \tag{1}$$

시그모이드 함수는 입력값이 커질수록 1로 수렴하고, 작아지면 0에 수렴 하는 것을 볼 수 있다. 하지만 이러한 특징 때문에 딥러닝 모델이 깊을수록 기울기가 사라지는 기울기 소실 문제가 발생한다.



(a) Sigmoid 함수 그래프
(a) Sigmoid function graph



(b) Sigmoid 미분 그래프
(b) Sigmoid differential graph
그림 4. Sigmoid 함수 및 미분 그래프
Fig. 4. Sigmoid function and differential graph

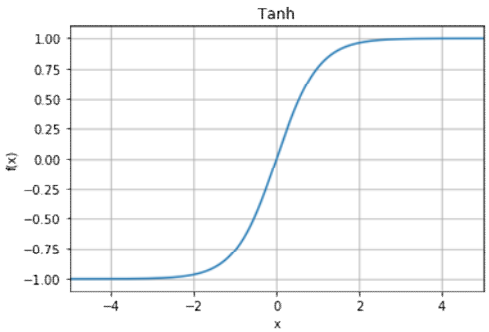
또한, 출력의 중심이 0이 아니라서 기울기 값이 모두 양수 혹은 모두 음수의 형태를 지녀 지그재그로 학습이 이루어져 비용과 효율 면에서 좋지 않다.

2.1.2 하이퍼볼릭 탄젠트 함수

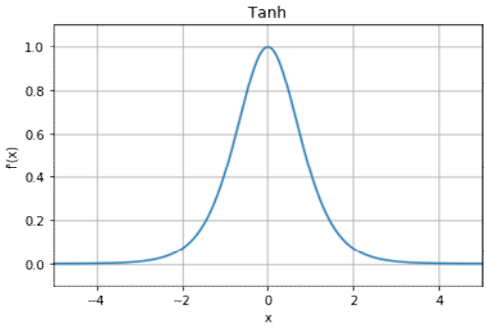
하이퍼볼릭 탄젠트 함수는 쌍곡선 함수이며 식 (2)과 같다.

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + x^{-x}} = 2\sigma(2x) - 1 \tag{2}$$

그림 5를 보면 출력의 중심이 0인 부분에서 시그모이드 함수의 단점을 해결하였다. 하지만 나머지 부분에서 시그모이드 함수와 같이 기울기 소실 문제가 발생하기 때문에 여전히 좋지 못한 함수이다.



(a) Tanh 함수 그래프
(a) Tanh function graph



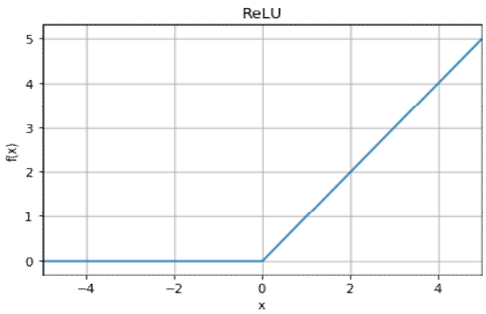
(b) Tanh 미분 그래프
(b) Tanh differential graph
그림 5. Tanh 함수 및 미분 그래프
Fig. 5. Tanh function and differential graph

2.1.3 ReLU(Rectified Linear Unit)

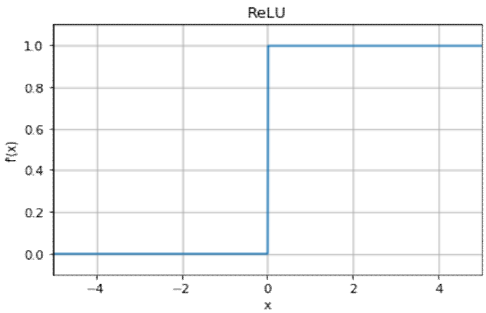
개선 선형 함수라고 하며 식 (3)과 같다. CNN에서 좋은 성능을 보였고, 현재 딥러닝에서 가장 많이 사용하는 활성 함수 중 하나이다.

$$f(x) = \begin{cases} 0 \ (x < 0) \\ x \ (x \geq 0) \end{cases} \tag{3}$$

기존의 시그모이드, 하이퍼볼릭 탄젠트 함수에 비해 학습 속도가 매우 빠르며 양수일 경우 0으로 수렴하지 않아 기울기 소실 문제를 해결하였다. 하지만 그림 6에서 보면 입력값이 0 또는 음수일 경우 기울기 값이 0이 되어 Dying ReLU 현상이 일어난다.



(a) ReLU 함수 그래프
(a) ReLU function graph



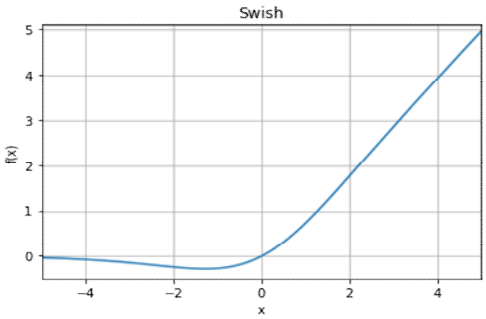
(b) ReLU 미분 그래프
(b) ReLU differential graph
그림 6. ReLU 함수 및 미분 그래프
Fig. 6. ReLU function and differential graph

2.1.4 Swish

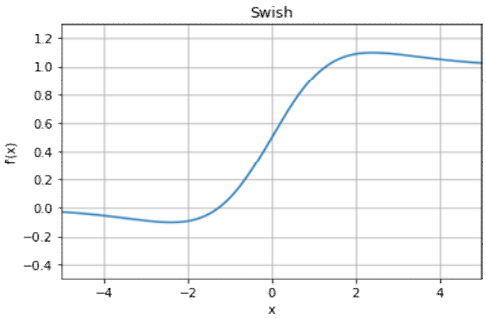
SiLU(Sigmoid Linear Unit)라고도 하며 식 (4)로 표현된다.

$$f(x) = \frac{x}{1 + e^{-x}} = x \cdot \ \sigma(x) \tag{4}$$

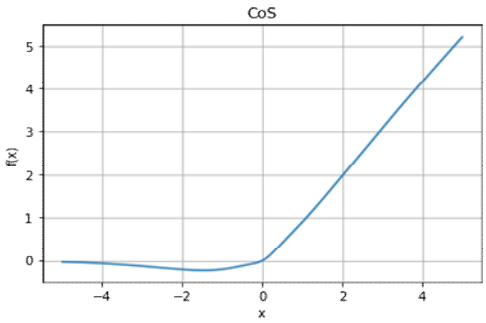
Sigmoid Function에 입력값을 곱한 함수로 그림 7과 같이 특이한 형태를 가진다. ReLU 보다 훨씬 부드러운 형태를 가지며 x = 0인 지점에서 미분이 가능하다. ReLU 및 다른 활성 함수를 대체하기 위해 만든 함수로 CIFAR-10 등의 데이터셋에서 좋은 성능을 가진다.



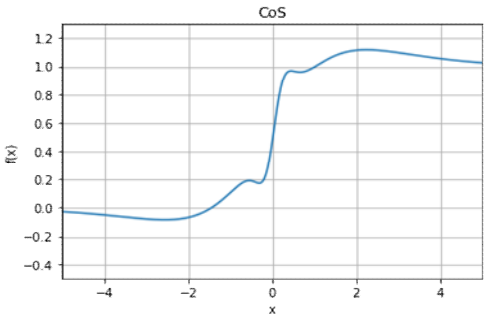
(a) Swish 함수 그래프
(a) Swish function graph



(b) Swish 미분 그래프
(b) Swish differential graph
그림 7. Swish 함수 및 미분 그래프
Fig. 7. Swish function and differential graph



(a) CoS 함수 그래프
(a) CoS function graph



(b) CoS 미분 그래프
(b) CoS differential graph
그림 8. CoS 함수 및 미분 그래프
Fig. 8. CoS function and differential graph

Swish[7]는 세 가지의 특성을 확인할 수 있다. 먼저, 시그모이드나 하이퍼볼릭 탄젠트 함수들과는 다르게 ReLU와 같이 위로 경계가 없이 무한의 값을 가지기 때문에 양의 값에서 어떤 높이로든 갈 수 있기에 학습하는 동안 0 근처에서 기울기 값이 포화되는 것을 막는다. 둘째, 부드러운 비단조성 곡선을 가지고 있어 일반화와 최적화에 중요한 역할을 한다. 또한, 네트워크 안에서 정보를 잘 흐르게 함으로, 초기값과 학습률에 덜 민감하다. 셋째, 아래로 음의 값에 대한 제한이 있기 때문에 강한 정규화 효과를 줄 수 있고 과적합을 줄일 수 있다. 또한, 작은 음수 입력값에도 음수 결과값을 만들 수 있어 표현력을 증가시키고 기울기 흐름을 개선한다.

III. 제안하는 방법

신경망의 모델에서 대부분 사용하는 활성 함수은 ReLU이다. 하지만 음의 영역에서 Dying ReLU 현상이 발생한다. 그러한 문제점을 Swish의 특성으로 해결하여 이미 좋은 성능을 보였다[8]-[10].

본 논문에서는 Swish의 형태를 기반으로 시그모이드 함수의 제곱 형태로 추가적인 부드러운 곡선을 조합하여 식 (5), 그림 8과 같이 시그모이드로 구성된 함수로 새로운 활성 함수(CoS, Consist of Sigmoid)를 제안한다.

$$f(x) = ((\sigma(5x) - 0.5)^2 + x) \cdot \sigma(x) \tag{5}$$

CoS는 추가된 시그모이드 함수의 제곱식을 x의 계수 5와 상수값 0.5으로 위치를 조절함으로써 Swish 모양과 비슷하게 만들어 Swish의 특성을 이용한다. 그리고 추가적인 부드러운 곡선을 가지고 있어 Swish보다 초기값과 학습률에 덜 민감하므로 일반화와 최적화에 강하다. 또한, 그림 7의 미분 그래프와 그림 8의 미분 그래프를 비교하면 추가적인 곡선으로 비단조성이 강조되어 기울기 변화에 따른 표현력 향상으로 네트워크의 정확도를 개선한다.

IV. 실험 결과

새롭게 제안한 활성 함수 CoS의 성능이 개선됨

을 보이기 위해 3가지의 활성 함수 ReLU, Swish, CoS를 이용하여 ShuffleNet v2, MobileNet v2, SqueezeNet에 적용했다. 데이터셋은 CIFAR-10을 사용하여 epoch=200을 통해 정확도와 손실값을 비교하였다.

그림 9, 10에서 정확도와 손실값을 보면 3가지 활성 함수 중 CoS 함수가 평균적인 손실값이 낮은 편에 속하고 정확도는 평균적으로 높은 것을 알 수 있다.

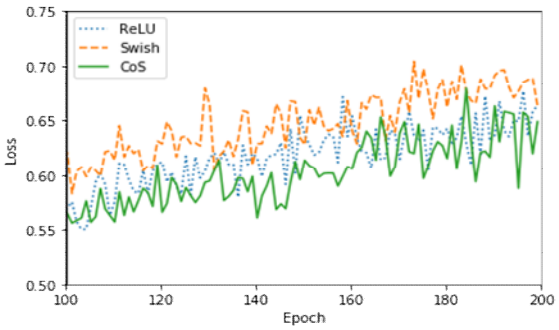


그림 9. ShuffleNet v2에 대한 Loss값 비교
Fig. 9. Comparison of loss value for ShuffleNet v2

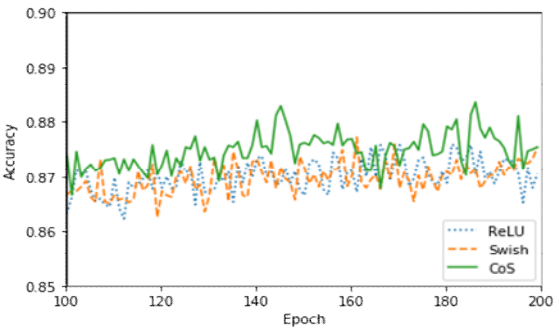


그림 10. ShuffleNet v2에 대한 Accuracy값 비교
Fig. 10. Comparison of accuracy value for ShuffleNet v2

표 1을 통해 수치적으로 보면 CoS 함수의 Top-1 loss는 0.4514로 가장 낮고 Top-1 accuracy는 ReLU보다 0.77% 높고 Swish보다 0.54% 높은 것을 확인할 수 있고 ShuffleNet v2 모델에서 CoS가 가장 좋은 결과를 볼 수 있다.

표 1. Activation Function에 따른 비교 (ShuffleNet v2)
Table 1. Comparison by activation function (ShuffleNet v2)

	ReLU	Swish	CoS
Time per epoch(s)	**36**	39	46
Top-1 loss	0.4673	0.4646	**0.4514**
Top-1 accuracy(%)	87.59	87.72	**88.36**

그림 11, 12에서 손실값을 보면 3가지 활성 함수 모두 비슷해 차이가 없고 평균적인 정확도에서 CoS가 높다는 것을 알 수 있다.

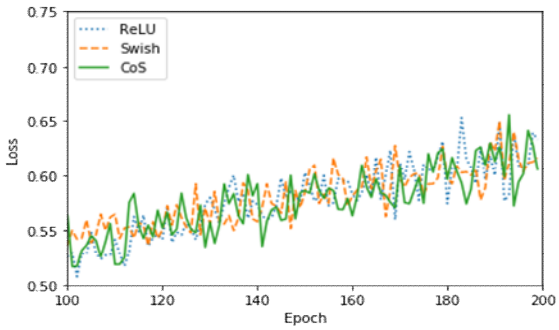


그림 11. MobileNet v2에 대한 Loss값 비교
Fig. 11. Comparison of loss value for MobileNet v2

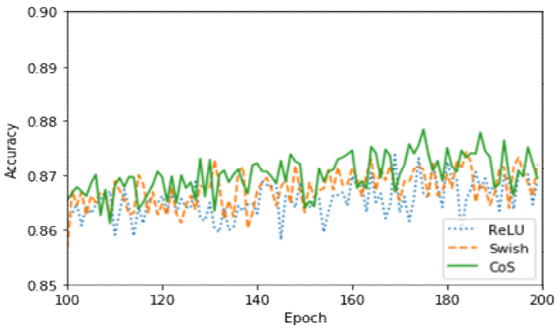


그림 12. MobileNet v2에 대한 각 Accuracy값 비교
Fig. 12. Comparison of accuracy value for MobileNet v2

표 2를 통해 수치적으로 보면 CoS 함수의 Top-1 loss는 0.4561로 Swish의 Top-1 loss보다 높지만 Top-1 accuracy는 ReLU보다 0.46%높고 Swish보다 0.41% 높은 것을 확인할 수 있다. MobileNet v2 모델에서 CoS 함수는 학습 시간과 손실값은 좋지 못하지만 가장 중요한 요소인 정확도는 CoS 함수가 가장 좋은 결과를 볼 수 있다.

표 2. Activation Function에 따른 비교 (MobileNet v2)
Table 2. Comparison by activation function (MobileNet v2)

	ReLU	Swish	CoS
Time per epoch(s)	**47**	55	72
Test loss	0.4667	**0.4415**	0.4561
Test accuracy(%)	87.39	87.44	**87.85**

그림 13, 14에서 손실값을 보면 3가지 활성 함수의 손실값은 SqueezeNet 모델에서 또한 비슷한 것을 볼 수 있고 CoS 함수의 정확도가 평균적으로 높은 것을 알 수 있다.

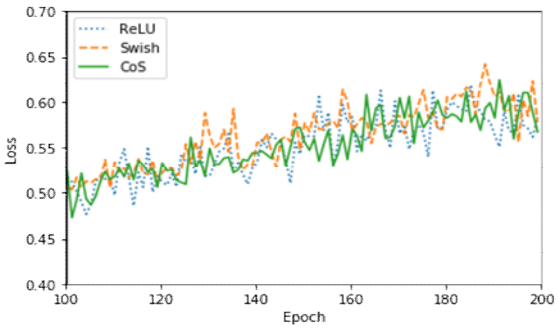


그림 13. SqueezeNet에 대한 각 Loss값 비교
Fig. 13. Compare loss value for SqueezeNet

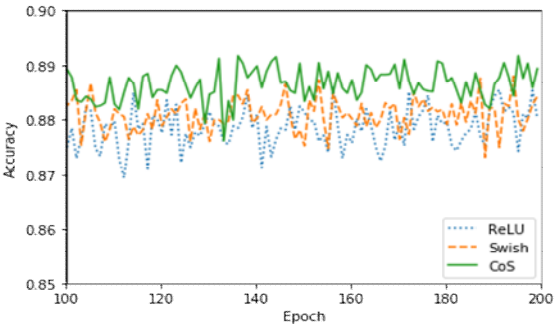


그림 14. SqueezeNet에 대한 각 Accuracy값 비교
Fig. 14. Compare accuracy value for SqueezeNet

표 3. Activation Function에 따른 비교 (SqueezeNet)
Table 3. Comparison by activation function (SqueezeNet)

	ReLU	Swish	CoS
Time per epoch(s)	**32**	36	45
Test loss	0.4193	**0.3810**	0.3927
Test accuracy(%)	88.58	88.80	**89.18**

표 3에서 수치적으로 보면 CoS 함수의 Top-1 loss는 0.3927로 Swish의 Top-1 loss보다 높지만 Top-1 accuracy는 ReLU보다 0.60% 높고 Swish보다 0.38% 높은 것을 확인할 수 있다. SqueezeNet 모델에서 또한 학습 시간과 손실값은 좋지 못하지만 가장 중요한 요소인 정확도는 CoS가 가장 좋은 결과를 볼 수 있다.

ShuffleNet v2, MobileNet v2, SqueezeNet 모델에서 활성 함수 ReLU, Swish, CoS를 비교하였을 때, 본 논문에서 제안하는 CoS 함수는 기존의 시그모이드 함수를 제곱의 형태로 추가적인 곡선으로 다른 활성 함수와 차별성을 두었고 부드러운 곡선의 추가로 일반화와 최적화의 기능을 향상시켰으며 계층 사이의 입출력 관계에 대한 복잡성을 높임에 따라 ShuffleNet v2에서 손실값, 정확도가 모두 개선된 결과를 보였다. MobileNet v2, SqueezeNet에서 손실값은 3가지 활성 함수 모두 비슷한 결과를 보여주어 차이가 없었지만 정확도의 성능 개선을 보였다.

3가지 모델 비교를 통해 CoS 함수는 제곱식이 포함되어 있어 학습 시간이 가장 느린 것을 알 수 있었다. 그렇지만 실제 이미지 인식에서는 학습이 완료된 모델을 가지고 적용하기 때문에 학습 시간보다 정확도가 중요하기 때문에 CoS 함수는 다른 함수들과 비교하여 개선된 성능을 보여준다.

그림 15는 MNIST 데이터셋을 사용하여 신경망 깊이에 따른 정확도를 활성 함수 ReLU, Swish, CoS를 적용하여 비교한 그래프이다. 본 논문에서 제안한 활성 함수(CoS)는 신경망이 깊어짐에 따라 Swish와 비슷한 정확도를 보이고 ReLU 보다 높은 정확도를 볼 수 있으며 깊은 신경망에 대한 정확도의 안정성을 확인할 수 있다.

무작위로 초기화된 단순한 신경망에서 출력 환경에 대한 Landscape를 그림 16를 통해 비교했다. ReLU를 보면 날카로운 변화를 보이고 Swish는 부드러운 변화를 보인다. 본 논문에서 새롭게 제안하는 CoS도 부드러운 변화를 보이기 때문에 ReLU 보다 부드러운 최적화 기능을 제공하고 손실을 줄이면서 신경망에 대한 일반화를 높일 수 있다.

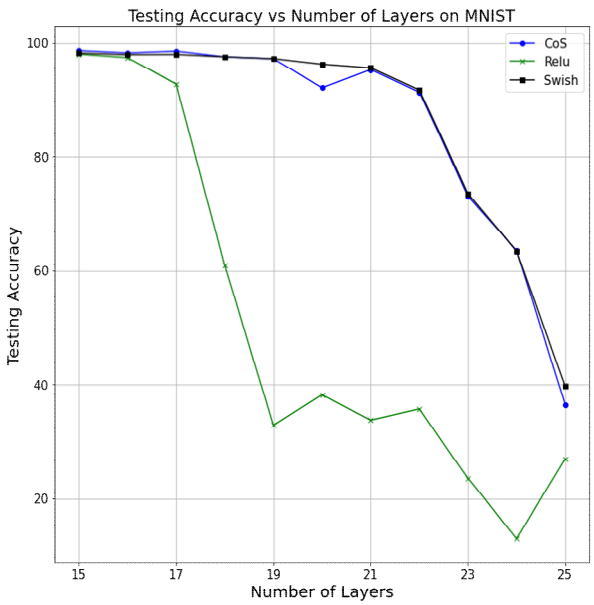
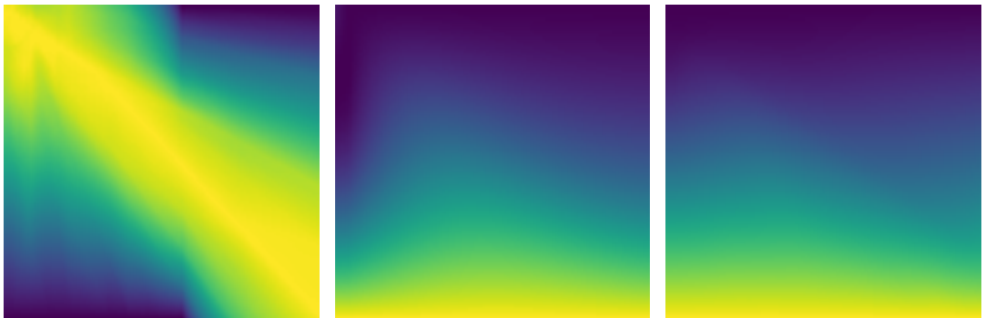


그림 15. Layer 깊이에 따른 정확도
Fig. 15. Accuracy according to layer depth



(a) ReLU (b) Swish (c) CoS

그림 16. ReLU, Swish and CoS landscape 비교

Fig. 16. Comparison of ReLU, Swish and CoS landscape

V. 결 론

본 논문에서 새로 제안된 활성 함수인 CoS는 Swish의 형태를 기반에 시그모이드 함수의 제곱 형태로 추가적인 부드러운 곡선을 조합하여 Swish의 특성을 가지면서 정확도의 개선을 보였다.

ShuffleNet v2, MobileNet v2, SqueezeNet에 총 epoch=200을 수행하였고 정확도는 ReLU 보다 0.46%~0.77%, Swish 보다는 0.38%~0.54%가 높아 정확도에 대한 성능 개선을 확인하였다.

하지만 제안하는 함수는 제곱식의 형태 추가로 인해 기존의 활성 함수인 ReLU, Swish보다 연산량이 많아 학습 시간이 가장 느린 것을 확인할 수 있었다. 추후 연구에서 연산량을 줄이기 위해 다른 함수를 조합하거나 제곱의 형태가 아닌 함수로 대체하여 함수를 수정하면 연산량을 줄여 학습 시간과 정확도 개선으로 향상된 결과를 얻을 것으로 예상된다.

References

[1] Md Foysal Haque and Dae-Seong Kang, "Deep Adversarial Residual Convolutional Neural Network for Image Generation and Classification", JAITC, Vol. 10, No. 1, pp. 111-120, Jul. 2020. https://doi.org/10.14801/jaitc.2020.10.1.111

[2] Su-Bin Park and Dae-Seong Kang, "A Study on H-CNN Based Pedestrian Detection Using LGP-FL and Hippocampal Structure", Journal of KIIT, Vol. 16, No. 12, pp. 75-83 Dec. 2018. https://doi.org/10.14801/jkiit.2018.16.12.75

[3] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition", ICLR 2015 Conference, San Diego USA, arXiv:1409.1556, May 2015.

[4] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, Massachusetts USA, pp. 1-9, Jun. 2015.

[5] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions", ICLR 2018 Conference, Vancouver, BC, Canada, arXiv: 1710.05941, Apr. 2018.

[6] S. Chae, S. Kim, and Mi. Park, "A Study on Novel Activation Function to Improve Conventional Activation Function of Deep Learning Algorithm", Autumn Annual Conference of IEIE, Gwangju Korea, pp. 920-922, Nov. 2018.

[7] P. Ramachandran, B. Zoph, and Q. V. Le, "Swish: a self-gated activation function", arXiv:1710. 05941v1, pp. 1-12, Oct. 2017.

[8] D. Misra, "Mish: A self regularized non-monotonic neural activation function", CoRR, pp. 1-13, Aug. 2019.

[9] H. H. Chieng, N. Wahid, P. Ong, and S. R. K. Perla, "Flatten-T Swish: a thresholded ReLU-

Swish-like activation function for deep learning", International Journal of Advances in Intelligent Informatics, Vol. 4, No. 2, pp. 76-85, Dec. 2018. https://doi.org/10.26555/ijain.v4i2.249.

[10] M. Tanaka, "Weighted sigmoid gate unit for an activation function of deep neural network", Pattern Recognition Letters, Vol. 135, pp. 354-359, Jul. 2020. https://doi.org/10.1016/j.patrec.2020.05.017

저자소개

한 준 (Jun Han)

2020년 2월 : 동아대학교 전자공학과 (공학사)
2020년 3월 ~ 현재 : 동아대학교 전자공학과 석사과정
관심분야 : 영상처리, 인공지능

강 대 성 (Dae-Seong Kang)

1994년 5월 : Texas A&M 대학교 전자공학과 (공학박사)
1995년 ~ 현재 : 동아대학교 전자공학과 교수
관심분야 : 영상처리