ISSN:1226-7244 (Print)
ISSN:2288-243X (Online)
논문번호 18-01-31
j.inst.Korean.electr.electron.eng.Vol.22,No.1,197~200,March 2018
http://dx.doi.org/10.7471/ikeee.2018.22.1.197

활성화 함수의 근사화를 통한 MLP 가속기 구현

MLP accelerator implementation by approximation of activation function

이 상 일 *, 최 세 진** ,이 광 엽**★

Sangil Lee*, Sejin Choi**, Kwangyeob Lee**★

Abstract

In this paper, sigmoid function, which is difficult to implement at hardware level and has a slow speed, is approximated by using PLAN. We use this as an activation function of MLP structure to reduce resource consumption and speed up. In this paper, we show that the proposed method maintains 95% accuracy in 5x5 size recognition and 1.83 times faster than GPGPU. We have found that even with similar resources as MLPA accelerators, we use more neurons and converge at higher accuracy and higher speed.

요 약

본 논문에서는 하드웨어레벨로 구현이 어렵고 속도가 느린 sigmoid 함수를 PLAN을 이용하여 근사치로 출력하였다. 이를 MLP 구조의 활성화 함수로 사용하여 자원소모를 줄이고 속도를 개선하고자 하였다. 본 논문에서 제안하는 방법은 5x5크기의 숫자 인식에 약 95%의 정확도를 유지하면서 GPGPU보다 약 1.83배의 빠른 속도를 보였다. 또한 MLPA가속기와 비슷한 자원을 사용함에도 더 많은 뉴런을 사용하여 높은 정확도에 빠른 속도로 수렴하는 것을 확인하였다.

Key words : Sigmoid function, PLAN, Machine Learning, MLP, ANN

*Dept. of Computer Eng., SeoKyeong University
★Corresponding author
E-mail: kylee@skuniv.ac.kr, Tel: +82-2-940-7745
※ Acknowledgment
Manuscript received Mar. 23, 2018; revised Mar. 29, 2018 ; accepted Mar. 29, 2018
This research was supported by the MOTIE(Ministry of Trade, Industry & Energy) (10080568) and KSRC(Korea Semiconductor Research Consortium) support program for the development of the future semiconductor device.

Ⅰ. 서론

인간이 사물을 인식하는 것처럼 컴퓨터가 사물을 사람처럼 인식하는 것에 관한 연구가 오래전부터 진행되어 왔으며 현재도 많은 알고리즘이 발표되고 있다. ANN(Artificial neural network)은 인간의 두뇌가 학습하는 과정을 토대로 만든 인공 신경망이다. 뇌를 구성하는 각 뉴런들의 연결관계를 모방하여 뇌의 정보처리 방식과 유사한 메커니즘을 구현하여 인공뉴런들을 대규모 네트워크 형태로 구성하는 구조이다.[1].

신경망의 알고리즘은 크게 지도학습과 비지도학습, 강화학습 등으로 구분할 수 있다. 지도학습은 학습 데이터에 대하여 출력에 대한 기댓값을

가지고 있으며 주어진 학습 데이터를 이용하여 입력과 출력사이의 대응관계가 잘 이루어지도록 네트워크를 연결한다. 비지도 학습은 별도의 학습 데이터 없이 데이터 자체를 분석하거나 군집하는 방법을 통하여 학습한다. 강화학습은 어떠한 문제를 해결할 때마다 보상을 주어 보상을 가장 많이 취하도록 학습하게 하는 학습방법이다. 딥러닝의 발전은 데이터 분류에 있어서 사람의 분류율과 유사하거나 더 좋은 성능을 보이는 등 실생활에 사람의 편의성을 향상시킬 수 있는 수준까지 도달하였다.[2]

본 논문에서는 지도 학습 방법을 통하여 5x5 사이즈의 숫자 인식을 테스트하는 MLP(Multi Layered Perceptron)구조를 하드웨어 레벨로 구현하였고 다른 ANN 구조와 자원사용량을 비교하였으며 GPGPU와 속도를 비교 하였다.

Ⅱ. 본론

1. MLP

MLP는 아래 그림 1처럼 입력층과 출력층 사이에 하나 이상의 중간층이 존재하는 구조로 구성되며 input layer에서 output layer로의 단방향 네트워크 모델이다.

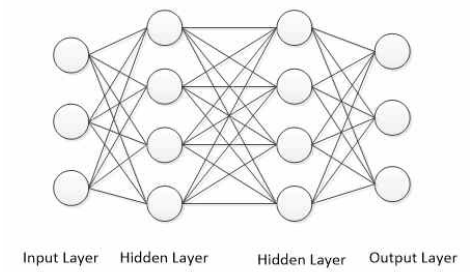

Fig. 1. structure of Multi layered perceptron
그림 1. Multi layered perceptron의 구조

MLP는 여러 개의 퍼셉트론이 하나의 네트워크를 이루는 것으로 구성된다. 하나의 퍼셉트론의 weight를 update하면서 학습하는 것은 쉬었다. 하지만 네트워크가 커지면서 복잡해져 학습이 복잡해진다. 이러한 weight를 최적화 하는 과정이 필요한데 이 과정이 back-propagation이다.

각 뉴런 하나는 그림 2처럼 동작한다. 각 수식과 동작방법에 대해서는 다음 파트에서 설명 한다.

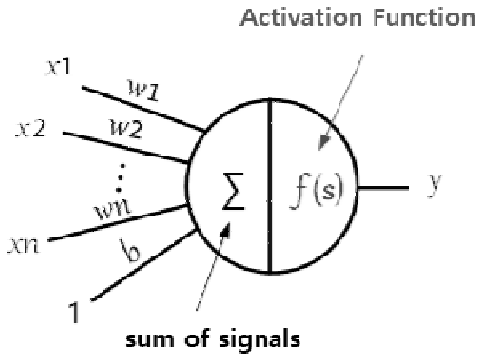

Fig. 2. operation of neuron
그림 2. 뉴런의 동작

2. Forward-propagation와 sigmoid function

모든 층의 경우 식(1)과 같이 weight와 입력 값들의 곱셈과 bias의 합으로 구성된다. weight는 뉴런의 영향력을 의미하며 bias는 신경망이 학습에 사용된 데이터에만 정확하게 판독하고 새로운 데이터에 대한 인식은 제대로 하지 못하는 문제를 방지하고자 일정한 값을 더하여 유연한 신경망을 구성하게 한다.

$$z = \sum_{i=0}^{n-1} x \cdot \quad w_i + b \tag{1}$$

식(1)의 결과 값인 z는 활성화 함수의 입력이 된다. 활성화 함수는 tanh, ReLU등 여러 종류가 있으나, 본 논문에서는 활성화함수로 sigmoid function을 사용하였다. 활성화함수는 연결된 다른 뉴런들로부터 전달된 값들을 식(2)의 연산을 진행한다.

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

제안하는 구조의 MLP의 forward-propagation 과정은 MAC(Multiply Accumulate) module로 구성된다. MAC는 weight와 입력을 곱한 후 결과값을 더해서 Activation Function으로 전달한다.

sigmoid function은 역치함수와 선형함수의 특징을 가지고 있는 비선형함수이다. 선형 함수를 사용하는 신경망의 경우 다층의 효과를 보기가 어렵다. 이것은 weight의 조정에 따라 동일한 효과를 갖는 단층 신경망으로 바뀔 수 있기 때문이다. 따라서 활성화함수가 비선형일 때 다층신경망의 장점이

발휘된다. sigmoid function은 입력 값이 아무리 작거나 크더라도 0과 1사이의 출력을 가진다. 이것은 신경망 내의 어느 한 뉴런이 신경망 전체의 동작을 좌지우지 하는 문제를 방지한다. [3] 또한 sigmoid function은 학습 시 가중치 업데이트를 위한 미분과정이 간단하다는 장점을 가지고 있다. [4] 식(2)에서 보이듯이 sigmoid function 의 경우 지수연산이 포함되어 있기 때문에 하드웨어 레벨로 구현하는 것이 복잡하며 속도가 느린 단점을 가진다. 이 같은 문제를 해결하기 위한 여러 방법들이 존재한다. 그 중 하나는 sigmoid function의 출력을 LUT(Look-Up Table)를 만들어 이를 통해 결정하는 방법이다. 하지만 하드웨어 자원이 많이 소모되며 시간이 오래 걸린다. 따라서 본 논문에서는 sigmoid function의 출력을 PLAN(Piecewise Linear Approximation of a Nonlinear function)을 이용하여 sigmoid function의 출력 값을 결정하였다. 제안하는 구조에서 아래 표 1과 같은 PLAN을 사용하여 하드웨어의 자원소모 및 시간 소모를 줄이고자 하였다.

Table 1. Implement PLAN behavior
표 1. PLAN 동작 구현

Condition	Operation
|X| >= 5	Y=1
2.375 =< |X| < 5	Y = 0.03125 * |X| + 0.84375
1 =< |X| < 2.375	Y = 0.125 * |X| + 0.625
0 =< |X| < 1	Y = 0.25 * |X| + 0.5
X < 0	Y = 1 - Y

3. Back-propagation

Back-Propagation은 실질적으로 학습이 이루어지는 과정이다. 원래 기대한 출력값과 실제 출력값 사이를 비교하는 cost function의 결과를 미분하는 과정을 통하여 weight를 업데이트하는 방식으로 학습을 진행한다. weight는 각 뉴런이 신경망에 얼마나 영향을 끼치는지를 결정하게 되므로 결국 학습은 적절한 weight를 찾는 것으로 진행된다. 역전파 과정을 통하여 계산된 델타 값, 이전 layer의 뉴런 값, learning rate를 모두 곱한 값이 기울기 값이 된다. learning rate는 학습의 속도를 결정하며 이를 통해 weight가 업데이트 되는 크기가 결정된다. Back-propagation은 에러를 역전파하는 과정이다.

4. MLP 하드웨어 가속기

제안하는 MLP구조는 그림 3과 같은 구조를 가지고 있으며 FSM(Finite State Machine)으로 동작한다. MLP 가속기는 control unit, 학습 알고리즘에 대한 연산을 수행하는 Forward-propagation unit, Back-propagation unit, weight generator, Data address Generator등으로 구성되며 Xilinx사에서 FPGA 개발환경을 위해 제공하는 CPU인 Micro Blaze를 사용하였다. 이 Micro Blaze는 학습 및 테스트 신호를 control Unit으로 전달하며 control Unit은 이 신호들을 이용하여 학습 및 테스트를 진행하여 가속기를 제어한다.

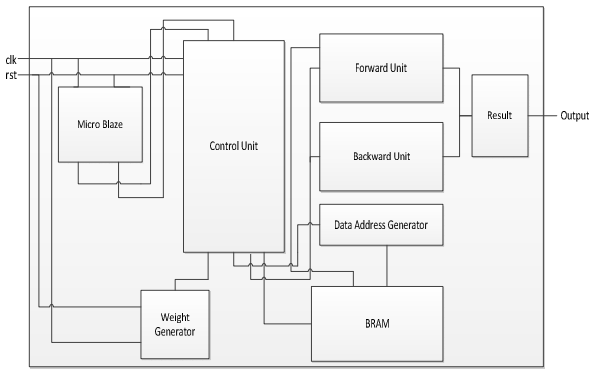

Fig.3. Structure of MLP hardware accelerator
그림 3. MLP 하드웨어 가속기의 구조

5. MLP 가속기의 제어

MLP 가속기의 제어 unit은 Micro Blaze로부터 학습 및 테스트신호를 입력 받으며 FSM형태로 동작한다. 학습 신호가 입력되었을 경우 설정되어 있는 Epoch만큼 Forward-propagation 연산과 Back-propagation연산을 반복수행하며 학습을 진행한다. 학습이 완료된 경우 test가능 신호를 출력하여 테스트가 가능함을 알려준다. 테스트는 입력데이터가 들어왔을 때 Forward-propagation을 통하여 데이터를 분류하는 것으로 진행된다. control unit은 FSM 형태로 동작한다.

III. 실험 및 결과

제안하는 MLP 하드웨어 가속기는 Xilinx사의 vc707 FPGA board를 사용하여 실험 및 검증하였다.

Table 2. Comparison of resource usage of proposed structure with MLPA
표 2. MLPA와 제안하는 구조의 자원 사용량 비교

Condition	MLPA	Proposed Method
Total Number Slice register	3,050	3,450
Number used as Flip Flops	2,003	2,630
Number used as Latches	1,041	820
Number of 4 input LUTs	3,100	3,283
Number of bonded IOBs	21	27

표2는 MLPA[5]와 제안하는 MLP구조의 자원사용량의 차이를 보여준다. 자원사용량은 큰 차이를 보이지 않지만 본 논문에서 사용하는 구조는 더 많은 뉴런을 사용하였으며 총 학습에 소요되는 epoch수가 줄어들었다. 기존의 가속기가 60000 epoch이상의 학습을 진행하며 90%이상의 정확도에 수렴하였다면 본 논문에서 제안하는 가속기는 100Mhz clock에 동작하며 3500 epoch만을 학습하여 약 95%의 정확도에 수렴하였으며 속도 또한 표 3과 같이 GPGPU보다 1.83배 빠른 속도를 보였다.

Table 3. Experiment result
표 3. 실험결과

	Processing Time(ns)
GPGPU	97
Proposed Method	53

IV 결론

본 논문에서는 숫자 인식을 위한 MLP하드웨어 가속기는 ANN의 학습 속도를 가속화하는데 목적을 두었으며 연산 수행속도를 측정하였다. 이는 GPGPU보다 빠른 속도로 동작하였으며 MLPA와의 비교에서 활성화 함수로 사용한 sigmoid function을 PLAN을 통해 근사화 하는 방법이 비슷한 자원사용량에도 더 적은 학습시간에 더 높은 정확도를 보였다. 따라서 제안하는 방법이 기존의 MLP가속기보다 더 좋은 성능을 보임을 확인할 수 있었다.

References

[1] Nara Im, “System identification of plane frames using an artificial neural network,” Donga University master degree, 2001.

[2] TaeHwan Kim “Investigations on representing latent space features for dynamic system control based on machine learning methods,” Korea University master degree, 2017.

[3] Youngsu Kim, “The prediction of surface settlement and tunnel’s behavior during tunnel excavation by using Artificial neural network,” Kyungpook National University master degree , 2007.

[4] EuiSun Lee, “Research on Safety Estimation of Amusement Devices Structure by Artificial Neural Network,” Konyang University master degree , 2011.

[5] Panca Mudji Rahardjo, Moch. Rif’an dan Nanang Sulistyanto, “The Implementation of Feedforward Backpropagation Algorithm for Digit handwritten Recognition in a Xilinx Spartan-3,” *Jurnal EECCIS,* vol.4, no.2, pp17-21, 2010.