스파이킹 신경망 추론을 위한 심층 신경망 가중치 변환

이정수*, 이지영*, 허준영*, 홍지만**
*한성대학교
**숭실대학교
love910321@naver.com, lee_ji0@naver.com, jyheo@hansung.ac.kr,
jiman@ssu.ac.kr

Deep Neural Network Weight Transformation for Spiking Neural Network Inference

Jungsoo Lee, Jiyoung Lee, Junyoung Heo, Jiman Hong
*School of Computer Engineering, Hansung University
**School of Computer Science & Engineering Soongsil University

요 약

스파이킹 신경망은 실제 뉴런의 작동원리를 적용한 신경망으로, 뉴런의 생물학적 메커니즘 덕분에 기존 신경망보다 훈련과 추론에 소모되는 전력이 적다. 최근 딥러닝 모델이 거대해지며 운용에 소모되는 비용 또한 기하급수적으로 증가하고 있으며, 이러한 이유로 스파이킹 신경망은 3세대 신경망으로 주목받으며 관련 연구가 활발히 진행되고 있다. 그러나 스파이킹 신경망을 산업에 적용하기 위해서는 많은 단계를 거쳐야 하며, 모델을 다시 훈련해야 하는 비용도 이에 포함된다. 본 논문에서는 기존의 훈련된 딥러닝 모델의 가중치를 추출하여 스파이킹 신경망의 가중치로 변환하는 방법을 통해 재훈련 비용을 최소화하는 방법을 제안한다. 또한, 변환된 가중치를 사용한 추론 결과와 기존 모델의 결과를 비교해 가중치 변환이 올바르게 작동함을 보인다.

1. 서 론

최근 딥러닝이 급속도로 발전함에 따라 대규모 딥러닝 모델들이 나타나고 있다. 모델의 크기가 커질수록 더욱 다양한 표현을 처리할 수 있다는 장점이 있지만, 이를 운용하기 위한 데이터나 전력 등 필요한 비용들도 동시에 증가하게 된다. 이를 해결하기 위해 다양한 방법들이 연구되고 있으며, 그중 하나가 3세대 신경망이라고 불리는 스파이크 신경망이다[1]. 스파이크 신경망은 기존 신경망의 단점인 전력 소모를 해결하고자, 뉴런의 메커니즘을 차용한 신경망이다. 인간의 뉴런은 작동하지 않을 때 휴지 상태(resting)로 존재하다가 자극이 들어올 때 활성화되는 특징이 있으며, 이러한 특징 덕분에 두뇌는 낮은 전력으로도 기억, 추론 등의 작업을 한 번에 처리할 수 있다[2]. 스파이킹 신경망도 이와 유사하게, 신경망 내부의 뉴런이 휴지 상태로 대기하다가 입력에 의해 뉴런의 전위가 변화하며, 전위가 임계치를 넘을 때 발화하여 1을 출력하게 된다[3].

본 논문에서는 파이토치(Pytorch) 프레임워크로 훈련한 모델을 높은 이식성을 가진 ONNX 모델로 배포한 뒤, 넨고(Nengo) 프레임워크로 구축한 스파이킹 신경망 모델에 가중치를 삽입하는 것으로 가중치 변환 방법을 제안한다. 가중치가 삽입된 스파이킹 모델로 추론한 결과를 기존 모델이 추론한 결과와 비교하는 것으로 가중치 변환으로 인한 정확도 손실을 측정한다.

이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No. 2019-0-00708, 뉴로모픽 아키텍처 기반 자율형 IoT 응용통합개발환경).

2. 관련 연구

2.1 Nengo

넨고(Nengo)는 대규모의 두뇌 신경망 모델을 시뮬레이션하기 위해 만들어진 파이썬 프레임워크이다. 넨고는 실제 뉴런의 스파이킹 메커니즘을 지원하므로 보다 실제 두뇌에 가까운 인공지능 신경망을 구현할 수 있으며, 심층 신경망 구축을 지원하는 nengo_dl 라이브러리를 포함한다. 사용자는 nengo_dl API에 텐서플로(tensorflow) 객체를 삽입하는 방법으로 빠르게 신경망을 구축할 수 있다.

2.2 Pytorch

파이토치(Pytorch)는 Facebook에서 개발한 파이썬 기반의 오픈소스 머신러닝 라이브러리이다. 파이토치는 파이썬 기반으로 구현되어있기 때문에 코드가 간결하며, GPU를 사용하기 때문에 월등한 속도로 행렬 연산이 가능하다. 널리 사용되는 딥러닝 라이브러리인 텐서플로(Tensorflow)와 비교하면, 텐서플로는 연산 전에 그래프를 정의하는 정적 그래프를 사용하기 때문에 훈련 도중에 상태를 변경하기 쉽지 않다. 파이토치는 모델이 고정되어있지 않고, 데이터를 넣는 것으로 그래프가 정의되는 동적 그래프를 사용하므로 보다 유연하게 코드를 작성할 수 있다는 장점이

있다. 또한, 프레임워크 자체적으로 ONNX 배포를 지원하므로 향후 높은 확장성을 기대할 수 있다.

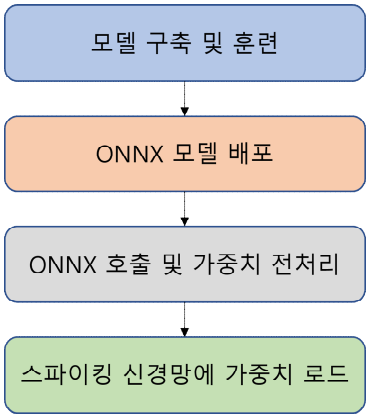
2.3 ONNX

ONNX (Open Neural Network eXchange)는 Facebook과 Microsoft가 개발한 신경망 모델의 표준 포맷이다. ONNX는 특정 프레임워크에서 생성한 모델을 타 프레임워크에서 사용 가능하게 하는 높은 이식성과 상호 운용성을 지원한다.

3. 가중치 변환

기존 신경망에서 스파이킹 신경망으로 가중치를 변환할 때 가중치만 직접 추출하는 대신, ONNX를 가중치 컨테이너로 사용한다. ONNX는 많은 딥러닝 프레임워크의 중간 표현 모델로 사용되고 있으며 가중치와 함께 레이어 구조 등의 모델 정보도 포함하고 있음으로 높은 확장성을 가지고 있다는 특징이 있다[4].

변환 과정은 그림 1과 같다. 기존 프레임워크에서 신경망 모델을 구축하고 훈련한 뒤 훈련된 모델을 ONNX 포맷으로 배포한다. 이후 넨고에서 ONNX 모델을 호출하여 담겨있는 가중치를 추출하여 스파이킹 신경망에 삽입하여 가중치 변환을 완료한다. 이때, 프레임워크에 따라서 사용하는 데이터 포맷과 가중치 구성이 다르므로 ONNX 모델을 호출한 뒤 가중치를 넨고에 맞게 전처리하는 과정을 거친다. 본 논문에서 사용한 ONNX는 가중치를 TensorProto 형식으로 저장하고, 넨고는 Numpy.float32로 저장하기 때문에 Numpy API를 사용하여 형식을 변환하였다. 또한 넨고는 가중치를 NPZ 형식으로 관리하기 때문에 변환한 데이터를 NPZ로 압축해서 넨고에 로드하였다.



변환 과정 (그림 1)

4. 실험

4.1 실험 환경

실험에 사용된 프레임워크의 버전은 표 1과 같다. 실험에는 각각 256개와 10개의 노드를 가진 전결합층(fully connected layer) 두 개로 구성된 신경망을 사용하였다. 데이터셋은 28x28의 손글씨 이미지인 MNIST를 사용하였으며, 훈련 데이터 60000장, 테스트 데이터 10000장으로 데이터 세트를 구성하였다. Optimizer는 Adam을 사용하였고, Learning rate는 0.0002로 설정하였다. 대상이 되는 파이토치 모델은 3 Epoch 동안 학습한 뒤 변환을 진행하였다.

프레임워크 버전 (표 1)

프레임워크	버 전
Pytorch	1.7.1
ONNX	1.7.0
Nengo	3.1.0
Nengo-DL	3.4.0
Tensorflow	2.4.0

4.2 실험 결과

실험 결과, 파이토치에서 91.32%의 정확도를 기록한 모델의 가중치를 변환하여 넨고의 스파이킹 네트워크로 추론했을 시, 87.67%의 정확도로 약 3.65%의 정확도 손실이 발생하였다. 이는 활성화 함수의 차이에서 기인하는데 가중치가 없는 기존 신경망의 활성화 함수와 달리, 스파이킹 신경망의 활성화 함수들에는 가중치가 존재하며 학습을 통해 최적의 발화 타이밍을 학습하게 된다[5]. 이 활성화 함수의 가중치는 기존 신경망으로 학습할 수 없기 때문에 초깃값을 사용하게 되고, 이로 인해 정확도 손실이 일부 발생하였다.

5. 결론

본 논문에서는 딥러닝 모델의 가중치를 스파이킹 신경망의 가중치로 변환하는 방법을 제안하였다. 딥러닝 모델 가중치를 저장하고 전달하는 컨테이너로 ONNX를 제안하였고, 가중치 변환 전후로 정확도의 변화를 측정하였다. 실험을 통해 약 3.65%, 기존 정확도의 0.04%에 해당하는 정확도 손실이 발생함을 확인하였고, 활성화 함수 가중치의 훈련 부재에서 오는 손실임을 알 수 있었다. 본 논문에서는 전결합층으로만 구성된 모델을 대상으로 실험을 진행하였는데 향후 보다 다양하고 거대한 모델에 대한 변환 기법을 제안한다.

참 고 문 헌

[1] K. I. Oh, S. E. Kim, and Y. H. Bae, “Trend of AI Neuromorphic Semiconductor Technology” Electronics and telecommunications trends Vol. 35, No. 3, 2018. pp. 76-84,

[2] Amirhossein Tavanaei, Masoud Ghodrati, Saeed Reza Kherad pisheh, Timothee Masquelier,Anthony Maida, “Deep Learning in Spiking Neural Networks”arXiv:1804.08150 v4, 2018

[3] W. J. Yu, Vu. Quoc. An, U. Y. Won, “Neuron-synapse devices and neuromorphic systems based on memristor” Communications of the Korean Institute of Information Scientists and Engineers Vol. 36, No. 6, pp. 2018. 71-80,

[4] 강대기, “딥러닝을 위한 인공신경망 표준 포맷 동향” TTA.Journal Vol.179. 2018. 85-90.

[5] Eric Hunsberger, and Chris Eliasmith, “Training Spiking Deep Networks for Neuromorphic Hardware” Arxiv, Neural and Evolutionary Computing, 2016