



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0151385
(43) 공개일자 2021년12월14일

(51) 국제특허분류(Int. Cl.)
G06N 3/063 (2006.01) G06F 17/16 (2006.01)
G06N 3/04 (2006.01)
(52) CPC특허분류
G06N 3/063 (2013.01)
G06F 17/16 (2013.01)
(21) 출원번호 10-2020-0068134
(22) 출원일자 2020년06월05일
심사청구일자 2020년06월05일

(71) 출원인
울산과학기술원
울산광역시 울주군 언양읍 유니스트길 50
(72) 발명자
이종은
울산광역시 울주군 언양읍 유니스트길 50
이수길
울산광역시 울주군 언양읍 유니스트길 50
(74) 대리인
특허법인태백

전체 청구항 수 : 총 12 항

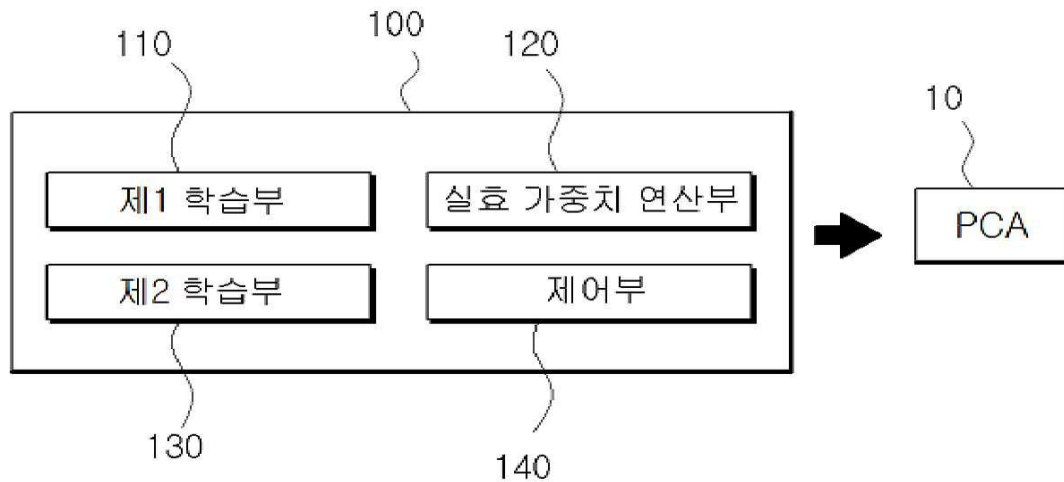
(54) 발명의 명칭 저항성 메모리 기반의 뉴럴 네트워크 가속기에 적용되는 연산 오차 보정 장치 및 그 방법

(57) 요약

본 발명에 따르면, 행렬-벡터 곱 연산을 수행하는 저항성 메모리(ReRAM) 기반의 뉴럴 네트워크 가속기에 적용되는 연산 오차 보정 장치에 있어서, 설정 입력 벡터와 임의 가중치 행렬을 포함한 학습 데이터를 이용하여, 임의 가중치 행렬의 입력에 대응하여 저항성 메모리의 전압 강하가 반영된 실패 가중치 행렬을 출력하기 위한 실패가

(뒷면에 계속)

대표도 - 도3



중치 예측 모델을 학습시키는 제1 학습부와, 연산 대상이 되는 제1 입력 벡터와 제1 가중치 행렬 중에서 제1 가중치 행렬을 실효 가중치 예측 모델에 입력시켜서 그에 대응된 실효 가중치 행렬을 도출하는 실효 가중치 연산부와, 타겟 뉴럴 네트워크에 제2 입력 벡터를 입력 값으로 적용하고 실효 가중치 행렬을 가중치 값으로 적용하여 출력 벡터를 도출하고 출력 벡터가 제2 입력 벡터와 제1 가중치 행렬 간 곱셈 결과를 추종하도록 가중치 값을 훈련시키는 제2 학습부, 및 제1 입력 벡터와 훈련된 가중치 행렬을 저항성 메모리에 매핑하여 행렬-벡터 곱 연산을 수행하는 제어부를 포함한다.

본 발명에 따르면, 저항성 메모리 기반의 뉴럴 네트워크 가속기에서 입력 벡터와 가중치 행렬 간의 행렬 곱 연산시에 저항성 메모리의 선저항으로 인한 행렬곱 계산 결과의 왜곡을 보정함으로써 저항성 메모리 기반의 뉴럴 네트워크 가속기에 대한 추론 성능을 향상시킬 수 있다.

(52) CPC특허분류

G06N 3/0454 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711104953
과제번호	2016M3A7B4910576
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	나노·소재기술개발(R&D)
연구과제명	재구성로직 아키텍처 기술개발
기 여 율	1/1
과제수행기관명	울산과학기술원
연구기간	2020.01.01 ~ 2020.12.31

명세서

청구범위

청구항 1

행렬-벡터 곱 연산을 수행하는 저항성 메모리(ReRAM) 기반의 뉴럴 네트워크 가속기에 적용되기 위한 연산 오차 보정 장치에 있어서,

설정 입력 벡터와 임의 가중치 행렬을 포함한 학습 데이터를 이용하여, 상기 임의 가중치 행렬의 입력에 대응하여 상기 저항성 메모리의 전압 강하가 반영된 실효 가중치 행렬을 출력하기 위한 실효 가중치 예측 모델을 학습시키는 제1 학습부;

연산 대상이 되는 제1 입력 벡터와 제1 가중치 행렬 중에서 상기 제1 가중치 행렬을 상기 실효 가중치 예측 모델에 입력시켜서 그에 대응된 실효 가중치 행렬을 도출하는 실효 가중치 연산부;

타겟 뉴럴 네트워크에 제2 입력 벡터를 입력 값으로 적용하고 상기 도출된 실효 가중치 행렬을 가중치 값으로 적용하여 출력 벡터를 도출하고 출력 벡터가 상기 제2 입력 벡터와 제1 가중치 행렬 간의 곱셈 결과를 추종하도록 타겟 뉴럴 네트워크의 가중치 값을 훈련시켜서 훈련된 가중치 행렬을 획득하는 제2 학습부; 및

상기 제1 입력 벡터와 상기 훈련된 가중치 행렬을 상기 저항성 메모리에 매핑하여 행렬-벡터 곱 연산을 수행하는 제어부를 포함하는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치.

청구항 2

청구항 1에 있어서,

상기 저항성 메모리는,

n 차원의 입력 벡터가 인가되는 n 개 비트 라인, $m \times n$ 크기의 가중치 행렬이 매핑되는 $n \times m$ 개 저항성 소자, 그리고 상기 행렬-벡터 곱 연산에 따른 m 차원의 출력 벡터를 도출하는 m 개 워드 라인을 포함하는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치.

청구항 3

청구항 1에 있어서,

상기 제1 학습부는,

상기 출력되는 실효 가중치 행렬이 상기 학습 데이터와 상기 저항성 메모리의 정보로부터 기 추정된 타겟 가중치 행렬을 추종하도록 상기 실효 가중치 예측 모델을 학습시키는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치.

청구항 4

청구항 3에 있어서,

상기 타겟 가중치 행렬은,

상기 저항성 메모리의 스펙 정보, 각 라인의 구성소재의 저항값, 해당 학습 데이터를 시뮬레이션에 적용하여 기 추정되는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치.

청구항 5

청구항 1에 있어서,

상기 실효 가중치 예측 모델은,

상기 저항성 메모리의 비트 라인 및 워드 라인에 각각 대응한 행 성분의 레이어 및 열 성분의 레이어를 조합하여 학습하는 RCN(Row-Column Network)으로 구현된 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치.

청구항 6

청구항 3에 있어서,

상기 제1 학습부는,

아래 수학적식에 의해 연산되는 평균 제곱 오차를 이용하여, 상기 실효 가중치 예측 모델에서 출력되는 실효 가중치 행렬(W_e)이 상기 타겟 가중치 행렬($\hat{W}_e$)을 추종하도록 학습시키는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치:

$$L = \frac{1}{N} \|\hat{W}_e - W_e\|_2^2$$

여기서, L은 평균 제곱 오차, $N = n \times m$ 을 나타낸다.

청구항 7

행렬-벡터 곱 연산을 수행하는 저항성 메모리(ReRAM) 기반의 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치를 이용한 연산 오차 보정 방법에 있어서, 설정 입력 벡터와 임의 가중치 행렬을 포함한 학습 데이터를 이용하여, 상기 임의 가중치 행렬의 입력에 대응하여 상기 저항성 메모리의 전압 강하가 반영된 실효 가중치 행렬을 출력하기 위한 실효 가중치 예측 모델을 학습시키는 단계;

연산 대상이 되는 제1 입력 벡터와 제1 가중치 행렬 중에서 상기 제1 가중치 행렬을 상기 실효 가중치 예측 모델에 입력시켜서 그에 대응된 실효 가중치 행렬을 도출하는 단계;

타겟 뉴럴 네트워크에 제2 입력 벡터를 입력 값으로 적용하고 상기 도출된 실효 가중치 행렬을 가중치 값으로 적용하여 출력 벡터를 도출하고 출력 벡터가 상기 제2 입력 벡터와 제1 가중치 행렬 간의 곱셈 결과를 추종하도록 타겟 뉴럴 네트워크의 가중치 값을 훈련시켜서 훈련된 가중치 행렬을 획득하는 단계; 및

상기 제1 입력 벡터와 상기 훈련된 가중치 행렬을 상기 저항성 메모리에 매핑하여 행렬-벡터 곱 연산을 수행하는 단계를 포함하는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 방법.

청구항 8

청구항 7에 있어서,

상기 저항성 메모리는,

n차원의 입력 벡터가 인가되는 n개 비트 라인, $m \times n$ 크기의 가중치 행렬이 매핑되는 $n \times m$ 개 저항성 소자, 그리고 상기 행렬-벡터 곱 연산에 따른 m차원의 출력 벡터를 도출하는 m개 워드 라인을 포함하는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 방법.

청구항 9

청구항 7에 있어서,

상기 실효 가중치 예측 모델을 학습시키는 단계는,

상기 출력되는 실효 가중치 행렬이 상기 학습 데이터와 상기 저항성 메모리의 정보로부터 기 추정된 타겟 가중치 행렬을 추종하도록 상기 실효 가중치 예측 모델을 학습시키는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 방법.

청구항 10

청구항 9에 있어서,

상기 타겟 가중치 행렬은,

상기 저항성 메모리의 스펙 정보, 각 라인의 구성소재의 저항값, 해당 훈련 데이터를 시뮬레이션에 적용하여 기 추정되는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 방법.

청구항 11

청구항 7에 있어서,

상기 실효 가중치 예측 모델은,

상기 저항성 메모리의 비트 라인 및 워드 라인에 각각 대응한 행 성분의 레이어 및 열 성분의 레이어를 조합하여 학습하는 RCN(Row-Column Network)으로 구현된 뉴럴 네트워크 가속기를 위한 연산 오차 보정 방법.

청구항 12

청구항 9에 있어서,

상기 실효 가중치 예측 모델을 학습시키는 단계는,

아래 수학적식에 의해 연산되는 평균 제곱 오차를 이용하여, 상기 예측 모델에서 출력되는 실효 가중치 행렬(W_e)이 상기 타겟 가중치 행렬($\hat{W}_e$)을 추종하도록 학습시키는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 방법:

$$L = \frac{1}{N} \|\hat{W}_e - W_e\|_2^2$$

여기서, L은 평균 제곱 오차, $N = n \times m$ 을 나타낸다.

발명의 설명

기술 분야

[0001] 본 발명은 저항성 메모리 기반의 뉴럴 네트워크 가속기에 적용되는 연산 오차 보정 장치 및 그 방법에 관한 것으로서, 보다 상세하게는 저항성 메모리의 선저항으로 인한 행렬곱 계산 결과의 왜곡을 보정하여 뉴럴 네트워크 가속기의 추론 성능을 향상시킬 수 있는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치 및 그 방법에 관한 것이다.

배경 기술

[0002] 뉴럴 네트워크 연산에서 대부분을 차지하는 핵심 작업은 행렬-벡터 곱(Matrix-Vector Multiplication) 연산으로, 이와 관련하여 저항성 메모리 어레이(ReRAM Crossbar Array; RCA)는 행렬-벡터 곱을 빠르고 효율적으로 구현할 수 있으며 뉴럴 네트워크 가속기를 위한 하드웨어로 사용되고 있다.

[0003] 이러한 저항성 메모리 어레이는 2차원 격자 형태로 저항성 소자를 배열한 구조를 가진다. 여기서, 곱하고자 하는 행렬의 값을 각 저항성 소자의 컨덕턴스 값(저항의 역수)으로 매핑하고, 곱하고자 하는 벡터의 값을 전압으로 걸어주면, $I=V/R$ 의 공식에 따라 전류 값의 형태로 행렬곱의 결과가 도출된다.

[0004] 그런데, 메모리 어레이는 선저항(wire resistance) 등의 영향으로 각 소자에 걸리는 전압이 기대값보다 낮아지는 전압 강하(IR drop)의 문제가 존재하며 이로 인하여 예측되는 전류 역시 원하는 값 보다 낮아져 행렬곱의 계산 결과에 왜곡이 나타난다. 나아가, 왜곡된 계산 결과는 뉴럴 네트워크의 추론 성능을 떨어뜨리는 요인이 된다.

[0005] 따라서, 저항성 메모리 어레이에 대한 전압 강하의 영향을 빠르게 예측함으로써 뉴럴 네트워크의 추정 정확도를 높일 수 있는 시스템이 요구된다.

[0006] 본 발명의 배경이 되는 기술은 한국공개특허 제2018-0042645호(2018.04.26 공개)에 개시되어 있다.

발명의 내용

해결하려는 과제

[0007] 본 발명은 저항성 메모리의 선저항으로 인한 행렬곱 계산 결과의 왜곡을 보정하여 뉴럴 네트워크 가속기의 추론

성능을 향상시킬 수 있는 저항성 메모리 기반의 뉴럴 네트워크 가속기에 적용되는 연산 오차 보정 장치 및 그 방법을 제공하는데 목적이 있다.

과제의 해결 수단

- [0008] 본 발명은, 행렬-벡터 곱 연산을 수행하는 저항성 메모리(ReRAM) 기반의 뉴럴 네트워크 가속기에 적용되기 위한 연산 오차 보정 장치에 있어서, 설정 입력 벡터와 임의 가중치 행렬을 포함한 학습 데이터를 이용하여, 상기 임의 가중치 행렬의 입력에 대응하여 상기 저항성 메모리의 전압 강하가 반영된 실효 가중치 행렬을 출력하기 위한 실효 가중치 예측 모델을 학습시키는 제1 학습부와, 연산 대상이 되는 제1 입력 벡터와 제1 가중치 행렬 중에서 상기 제1 가중치 행렬을 상기 실효 가중치 예측 모델에 입력시켜서 그에 대응된 실효 가중치 행렬을 도출하는 실효 가중치 연산부와, 타겟 뉴럴 네트워크에 제2 입력 벡터를 입력 값으로 적용하고 상기 도출된 실효 가중치 행렬을 가중치 값으로 적용하여 출력 벡터를 도출하고 출력 벡터가 상기 제2 입력 벡터와 제1 가중치 행렬 간의 곱셈 결과를 추종하도록 타겟 뉴럴 네트워크의 가중치 값을 훈련시켜서 훈련된 가중치 행렬을 획득하는 제2 학습부, 및 상기 제1 입력 벡터와 상기 훈련된 가중치 행렬을 상기 저항성 메모리에 매핑하여 행렬-벡터 곱 연산을 수행하는 제어부를 포함하는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치를 제공한다.
- [0009] 또한, 상기 저항성 메모리는, n차원의 입력 벡터가 인가되는 n개 비트 라인, $m \times n$ 크기의 가중치 행렬이 매핑되는 $n \times m$ 개 저항성 소자, 그리고 상기 행렬-벡터 곱 연산에 따른 m차원의 출력 벡터를 도출하는 m개 워드 라인을 포함할 수 있다.
- [0010] 또한, 상기 제1 학습부는, 상기 출력되는 실효 가중치 행렬이 상기 학습 데이터와 상기 저항성 메모리의 정보로부터 기 추정된 타겟 가중치 행렬을 추종하도록 상기 실효 가중치 예측 모델을 학습시킬 수 있다.
- [0011] 또한, 상기 타겟 가중치 행렬은, 저항성 메모리의 스펙 정보, 각 라인의 구성소재의 저항값, 해당 학습 데이터를 시뮬레이션에 적용하여 기 추정될 수 있다.
- [0012] 또한, 상기 실효 가중치 예측 모델은, 상기 저항성 메모리의 비트 라인 및 워드 라인에 각각 대응한 행 성분의 레이어 및 열 성분의 레이어를 조합하여 학습하는 RCN(Row-Column Network)으로 구현될 수 있다.
- [0013] 또한, 상기 제1 학습부는, 아래 수식식에 의해 연산되는 평균 제곱 오차를 이용하여, 상기 실효 가중치 예측 모델에서 출력되는 실효 가중치 행렬(W_e)이 상기 타겟 가중치 행렬($\hat{W}_e$)을 추종하도록 학습시킬 수 있다.

$$L = \frac{1}{N} \|\hat{W}_e - W_e\|_2^2$$

[0014]

여기서, L은 평균 제곱 오차, $N = n \times m$ 을 나타낸다.

[0015]

[0016]

그리고, 본 발명은, 행렬-벡터 곱 연산을 수행하는 저항성 메모리(ReRAM) 기반의 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치를 이용한 연산 오차 보정 방법에 있어서, 설정 입력 벡터와 임의 가중치 행렬을 포함한 학습 데이터를 이용하여, 임의 가중치 행렬의 입력에 대응하여 저항성 메모리의 전압 강하가 반영된 실효 가중치 행렬을 출력하기 위한 실효 가중치 예측 모델을 학습시키는 단계와, 연산 대상이 되는 제1 입력 벡터와 제1 가중치 행렬 중에서 상기 제1 가중치 행렬을 상기 실효 가중치 예측 모델에 입력시켜서 그에 대응된 실효 가중치 행렬을 도출하는 단계와, 타겟 뉴럴 네트워크에 제2 입력 벡터를 입력 값으로 적용하고 상기 도출된 실효 가중치 행렬을 가중치 값으로 적용하여 출력 벡터를 도출하고 출력 벡터가 상기 제2 입력 벡터와 제1 가중치 행렬 간의 곱셈 결과를 추종하도록 타겟 뉴럴 네트워크의 가중치 값을 훈련시켜서 훈련된 가중치 행렬을 획득하는 단계, 및 상기 제1 입력 벡터와 상기 훈련된 가중치 행렬을 상기 저항성 메모리에 매핑하여 행렬-벡터 곱 연산을 수행하는 단계를 포함하는 뉴럴 네트워크 가속기를 위한 연산 오차 보정 방법을 제공한다.

발명의 효과

[0017]

본 발명에 따르면, 저항성 메모리 기반의 뉴럴 네트워크 가속기에서 입력 벡터와 가중치 행렬 간의 행렬 곱 연산 시에 저항성 메모리의 선저항으로 인한 행렬곱 계산 결과의 왜곡을 보정함으로써 저항성 메모리 기반의 뉴럴 네트워크 가속기에 대한 추론 성능을 향상시킬 수 있다.

도면의 간단한 설명

- [0018] 도 1은 본 발명의 실시예에 적용되는 저항성 메모리 기반의 뉴럴 네트워크 가속기의 하드웨어 구조를 나타낸 도면이다.
- 도 2는 본 발명의 실시예에서 실패 가중치를 이용하여 MVM 연산 결과의 왜곡을 보상하는 개념을 설명한 도면이다.
- 도 3은 도 1에 도시된 저항성 메모리 기반의 뉴럴 네트워크 가속기에 적용되는 연산 오차 보정 장치의 구성을 나타낸 도면이다.
- 도 4는 도 2를 이용한 연산 오차 보정 방법을 설명하는 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0019] 그러면 첨부한 도면을 참고로 하여 본 발명의 실시 예에 대하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시 예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.
- [0020] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결"되어 있는 경우도 포함한다. 또한 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다.
- [0021] 도 1은 본 발명의 실시예에 적용되는 저항성 메모리 기반의 뉴럴 네트워크 가속기의 하드웨어 구조를 나타낸 도면이다.
- [0022] 도 1에 도시된 저항성 메모리(10)는 크로스바 어레이 구조의 저항 변화 메모리(ReRAM Crossbar Arrays, RCA)로서 MVM 컴퓨팅을 수행하는 수동형 RCA(passive RCA)이며 MVM 연산을 위한 뉴럴 네트워크 가속기 하드웨어로 동작한다.
- [0023] 즉, 저항성 메모리(10)는 뉴럴 네트워크의 연산에서 가장 많은 비중을 차지하는 MVM 연산을 위한 뉴럴 네트워크 하드웨어로 적용된다. RCA 구조는 MVM 연산 시 속도, 전력, 공간 면에서 큰 장점을 갖는다.
- [0024] 도 1에 나타난 것과 같이, 저항성 메모리(10)는 n 차원의 입력 벡터 $[x_1, x_2, \dots, x_n]^T$ 가 인가되기 위한 n 개 비트 라인(L1), $m \times n$ 크기의 가중치 행렬(W)이 매핑되는 $n \times m$ 개 저항성 소자(R), 그리고 행렬-벡터 곱 연산에 따른 m 차원의 출력 벡터 $[y_1, x_2, \dots, y_m]$ 를 도출하는 m 개 워드 라인(L2)을 포함한다.
- [0025] 저항성 메모리(10)에 인가되는 입력 벡터를 X , 가중치 행렬을 W , 출력 벡터를 Y 라고 할 때, $Y=WX$ 가 성립된다. 따라서, Y 는 MVM(행렬-벡터 곱) 연산의 출력을 나타낸다.
- [0026] 도 1에서 저항성 소자(R) 각각에 매핑된 G_{ij} 값은 저항(resistance)의 역수인 컨덕턴스(conductance; 전도도) 값에 해당한다. $i=\{1, \dots, n\}$, $j=\{1, \dots, m\}$ 이다.
- [0027] 상술한 바와 같은 저항성 메모리(10)의 하드웨어 구조를 이용한 벡터-행렬 곱 연산은 다음과 같은 방법으로 이루어진다.
- [0028] 우선, MVM 연산 대상이 되는 행렬 W 와 벡터 X 가 있다고 할 때, 벡터 X 의 각 원소($x_1, x_2, \dots, x_n$)를 n 개 비트 라인(L1)에 걸리는 전압 값으로 각각 매핑하고, 행렬 W 의 각 원소를 각 저항성 소자(R)의 컨덕턴스 값으로 각각 매핑함으로써, 아래 수학적 1의 곱셈 연산을 수행할 수 있다.

수학적 1

[0029] $W \cdot X = Y$

- [0030] 구체적으로 전압값과 컨덕턴스 간의 곱셈 연산이 수학적 2와 같은 방법으로 이루어지면서, 저항성 메모리(10)의

m개 워드 라인(L2)을 통하여 m차원의 전류 값($y_1, y_2, \dots, y_m$)이 최종 출력된다.

수학식 2

$$\begin{bmatrix} G_{11} & G_{21} & \cdots & G_{n1} \\ G_{12} & G_{22} & \cdots & G_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ G_{1m} & G_{2m} & \cdots & G_{nm} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = [y_1 \quad y_2 \quad \cdots \quad y_m]$$

[0031]

물론, 도 1에 따른 컨덕턴스 행렬(G)과 수학식 1과 2의 가중치 행렬(W) 간 관계는 아래와 같이 표현될 수 있다.

수학식 3

$$G = \begin{bmatrix} G_{11} & G_{12} & \cdots & G_{1m} \\ G_{21} & G_{22} & \cdots & G_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ G_{n1} & G_{n2} & \cdots & G_{nm} \end{bmatrix} = W^T$$

[0033]

이와 같이, 저항성 메모리(10)는 n개 비트 라인(L1)에 입력되는 n차원의 입력 벡터(X)와, 각 저항성 소자(R)의 컨덕턴스 값으로 결정되는 가중치 행렬(W) 간에 행렬-벡터 곱(MVM) 연산을 수행하고, 이를 통해 m개 워드 라인(L2)을 통해 m차원의 출력 벡터(Y)를 출력한다.

[0034]

그런데, 이러한 저항성 메모리(10)는 하드웨어 구조상 라인의 선저항(wire resistance)으로 인해 IR 드롭(전압 강하)이 존재하며 이는 실제 기대한 곱셈 결과(참값; Y) 보다 낮은 크기의 출력 값(Y')을 유발한다.

[0035]

이러한 선저항에 의한 IR 드롭 문제는 행렬-벡터 곱(MVM) 연산 결과에 오차(왜곡)를 유발하고 이러한 연산 오차는 뉴럴 네트워크를 통한 분석 및 추정 성능을 저하시키는 요인이 된다.

[0036]

더욱이, 전압 강하의 영향은 $n \times m$ 개의 모든 저항성 소자(R)에 동일한 크기로 나타나지 않는데, 특히 입/출력 포트에서 먼 소자일수록 크게 나타나며, 선저항의 크기와 주변 소자의 저항 값에도 영향을 받는다. 또한, SPICE 시뮬레이션 방법은 전압 강하의 영향을 정확하게 시뮬레이션할 수는 있으나 시간이 매우 많이 소요되는 단점이 있어, 실시간 빠른 연산을 필요로 하는 뉴럴 네트워크의 실제 연산에 적용하는 것은 거의 불가능하다. 따라서, 전압 강하의 영향을 예측할 수 있는 빠르고 정확한 새로운 모델이 요구된다.

[0037]

도 2는 본 발명의 실시예에서 실효 가중치를 이용하여 MVM 연산 결과의 왜곡을 보상하는 개념을 설명한 도면이다.

[0038]

도 2의 (a)는 저항성 메모리(10) 내에 MVM 연산 왜곡이 존재하지 않는 이상적인 경우를 나타낸다. 이 경우 입력 벡터 X와 가중치 행렬 W 간의 곱셈 연산 결과, 실제 기대한 참값(Y)이 도출된다.

[0039]

하지만, 도 2의 (b)는 선저항으로 인한 MVM 연산 왜곡이 존재한 상황을 나타낸다. 이 경우 벡터 X와 행렬 W의 곱셈 연산 결과, 전압 강하의 영향으로 인하여 참값(Y)보다 작은 왜곡된 결과(Y')를 도출한 것을 알 수 있다. 이처럼 저항성 메모리(10) 내에 MVM 연산 왜곡이 존재할 경우 뉴럴 네트워크(DNN)의 추론 성능이 저하된다.

[0040]

도 2의 (b)의 결과는 $Y' = g(W, X) \neq WX$ 로 요약될 수 있다. 이러한 왜곡된 연산 결과 Y'를 도출하는 함수인 g는 SPICE와 같은 시뮬레이션을 통해 추정 가능하지만 SPICE 시뮬레이션의 경우 결과값을 얻는 데까지 상당한 시간이 요구된다.

[0041]

본 발명의 실시예는 함수 g를 구하는 방법 대신, 도 2의 (c)와 같이, 별도의 예측 모델인 $h(W) = W_e$ 을 사용하여, 실제 연산하고자 한 가중치(W)로부터 실효 가중치(W_{EFF} ; 이하, W_e)를 예측한다.

[0042]

- [0043] 이때, 실효 가중치(W_e)는 원래의 W 값이 아닌 저항성 메모리(10)의 선저항에 따른 전압 강하의 영향이 작용한 실효치로서, 왜곡된 연산 결과는 실효 가중치와 입력 간의 곱($Y' = W_e \cdot X$)으로 표현될 수 있다. 본 발명의 실시예는 이와 같이 실제 왜곡되는 계산 결과에 맞게 타겟 뉴럴 네트워크를 훈련한다.
- [0044] 이를 위해, 우선 실효 가중치 예측 모델을 구축하여 이를 기반으로 원래 사용할 가중치 행렬(W)로부터 왜곡된 MVM 영향이 반영된 실효 가중치 행렬(W_e)를 도출한 다음 도출된 실효 가중치 행렬(W_e)을 행렬-벡터 곱 연산을 수행하는 타겟 뉴럴 네트워크의 훈련에 적용하여 재훈련한다.
- [0045] 재훈련 과정에서는 타겟 뉴럴 네트워크에서 실효 가중치 행렬(W_e)과 임의 입력 벡터(X) 간 수행한 행렬-벡터 곱 연산 결과($Y' = W_e X$)가 실제 도출되어야 하는 목표 값($Y = WX$)을 추종하도록 가중치를 재조정하는 훈련을 수행함으로써, 훈련된 가중치 행렬(W')이 새롭게 얻어진다. 새롭게 구해진 가중치 행렬(W')을 원래 사용할 가중치 행렬(W) 대신 저항성 메모리(10) 내 저항성 소자의 각 컨덕턴스 값으로 적용할 경우에 전압 드롭의 영향이 상쇄된 정확한 출력값 $Y(W' \cdot X = Y)$ 를 얻을 수 있다.
- [0046] 이와 같이, 본 발명의 실시예의 경우, MVM 연산 대상이 되는 벡터(X)와 행렬(W)이 있다고 할 때, 미리 학습된 예측 모델을 기반으로 해당 저항성 메모리(10)의 선저항에 의한 전압 강하가 반영된 실효 가중치 행렬(W_e)을 먼저 예측한다. 그리고, 이 실효 가중치 행렬(W_e)을 타겟 뉴럴 네트워크에서 재학습함으로써 조정된(훈련된) 가중치 행렬(W')을 획득하고 이를 실제 저항성 메모리(10)의 MVM 연산 시 원래의 가중치 행렬(W) 대신에 업데이트하여 사용한다.
- [0047] 업데이트된 가중치 행렬을 저항성 메모리 내의 각 소자의 컨덕턴스 값으로 사용함으로써, 왜곡이 보정된 참값(Y) 또는 그에 가까운 정확도 높은 추론 결과를 도출할 수 있다. 이러한 방법으로 선저항으로 인해 발생하는 뉴럴 네트워크의 추론 성능 저하 문제를 효과적으로 해결할 수 있다.
- [0048] 즉, 본 발명은 실제 저항성 메모리(10) 하드웨어로 구현된 뉴럴 네트워크 가속기의 연산 오차를 최소화하여, 이로부터 해당 뉴럴 네트워크의 추정 성능을 높일 수 있다. 만일, 해당 뉴럴 네트워크(DNN)가 이미지 내 객체를 분류(예: 사람, 자동차 등의 분류)하는 알고리즘인 경우, 본 발명의 실시예를 적용할 경우, 행렬곱(MVM) 연산 오류를 줄임으로써, 객체 분류의 성능 및 정확도를 높일 수 있게 된다.
- [0049] 다음은 본 발명의 실시예에 따른 저항성 메모리 기반의 뉴럴 네트워크 가속기에 적용되는 연산 오차 보정 장치의 구성을 더욱 상세히 설명한다.
- [0050] 도 3은 도 1에 도시된 저항성 메모리 기반의 뉴럴 네트워크 가속기에 적용되는 연산 오차 보정 장치의 구성을 나타낸 도면이다.
- [0051] 도 3에 나타난 것과 같이, 저항성 메모리 기반의 뉴럴 네트워크 가속기를 위한 연산 오차 보정 장치(100)는 제1 학습부(110), 실효 가중치 연산부(120), 제2 학습부(130), 제어부(140)를 포함한다.
- [0052] 제1 학습부(110)는 설정 입력 벡터와 임의 가중치 행렬을 포함한 학습 데이터를 이용하여, 임의 가중치 행렬의 입력에 대응하여 저항성 메모리의 전압 강하가 반영된 실효 가중치 행렬을 출력하기 위한 실효 가중치 예측 모델을 학습시킨다.
- [0053] 예를 들어, 입력 벡터와 임의 가중치 행렬은 1 또는 0의 이진 데이터의 조합으로 구성될 수 있으며, 임의 가중치 행렬은 별도의 생성기를 이용하여 임의로 랜덤하게 생성될 수 있다.
- [0054] 본 발명의 실시예는 예측 모델의 신뢰도를 높이기 위하여 다수의 훈련 데이터 세트를 구성하여 학습에 활용한다.
- [0055] 물론 이를 위해, 제1 학습부(110)는 해당 학습 데이터와 저항성 메모리(10)의 정보로부터 기 추정된 타겟 가중치 행렬 W_e 을 수집하고, 이를 해당 학습 데이터와 세트로 구성하여 사용한다.
- [0056] 타겟 가중치 행렬은 기존의 SPICE(Simulation Program with Integrated Circuit Emphasis) 시뮬레이션을 통해 사전에 추정될 수 있다. 시뮬레이션을 위한 파라미터 값으로 저항성 메모리의 스펙 정보(기하학 구조, 크기, 속성 등), 해당 훈련 데이터 등이 활용될 수 있다. 여기서 각 라인을 구성하는 금속 소재의 저항값(선저항)이 스

팩 정보에 포함될 수도 있다. 따라서, 타겟 가중치 행렬 W_e 은 저항성 메모리(10)에서 전압 강하의 영향이 반영된 유효 가중치 행렬과 같은 개념에 해당하며 시뮬레이션에 의해 획득된 값에 해당한다.

[0057] 정확한 실효 가중치를 알기 위해서는 SPICE 시뮬레이션을 사용해야 하지만 훈련 과정에 적용하기에는 너무 느려서 사용할 수 없다. 대신 본 발명의 실시예는 주어진 가중치(W)로부터 실효 가중치(W_e)를 예측하는 모델의 학습에 필요한 데이터(타겟 가중치 행렬)를 생성할 때와, 최종적으로 추론 성능을 검증(Validation)할 때만 SPICE 시뮬레이션을 사용한다. 이는 학습에는 수천번 이상의 시뮬레이션이 필요한 반면에, 검증에는 수백 번 이내의 수준으로 충분히 가능한 시간 범위 내에 사용할 수 있기 때문이다.

[0058] 제1 학습부(110)는 임의 가중치 행렬을 입력 변수로 하고 그에 대응되는 실효 가중치 행렬을 출력으로 하는 실효 가중치 예측 모델을 구성하고 이를 기반으로 예측 모델을 학습시킨다.

[0059] 여기서, 제1 학습부(110)는 실효 가중치 예측 모델로부터 출력되는 실효 가중치 행렬이 타겟 가중치 행렬을 추종하도록 실효 가중치 예측 모델을 학습시킨다.

[0060] 여기서, 저항성 메모리(10)에서 행렬곱 왜곡의 영향이 가로(입력) 방향과 세로(출력) 방향으로 모두 존재하기 때문에, 본 발명의 실시예의 경우 두 방향의 영향을 나누어 예측하는 모델을 사용한다.

[0061] 구체적으로, 본 발명의 실시예에 따른 실효 가중치 예측 모델은 저항성 메모리(10)의 비트 라인 및 워드 라인에 각각 대응한 행(row) 성분의 레이어 및 열 성분(column)의 레이어를 조합하여 학습하는 RCN(Row-Column Network)으로 구현될 수 있다.

[0062] RCN은 비선형 모델로, 복수의 뉴럴 네트워크 레이어(neural network layer)의 적층을 통하여 구성된다. 전압 강하(IR drop)는 행 방향과 열 방향에서 모두 존재하므로, 행방향과 열방향 모델을 결합하여 구성하는 것이 바람직하다. 따라서, 우선, 행방향과 열방향의 레이어를 개별 정의한 다음, 이들을 결합함으로써 RCN을 구성할 수 있다. 그 구체적인 실시예는 다음과 같다.

[0063] 가로 방향의 영향을 예측하는 모델(PL_{row})은 수학적 식 4와 같이 표현될 수 있으며, 주어진 행렬을 행 별로 독립적인 선형 회귀 모델로 병렬 처리한다. 이에 따라, 일반적인 선형 레이어(linear layer)를 병렬로 연결한 형태인 병렬 선형 레이어(parallel linear layer)가 된다.

수학적 식 4

$$[0064] \quad PL_{row} : X \rightarrow Y \text{ where } Y^{[i]} = f(X^{[i]} \cdot R_i + \vec{b}_i)$$

[0065] 여기서, $X^{[i]}$ 는 설정 입력 벡터에 대한 i번째 행 성분, $f(\cdot)$ 는 비선형 활성화 함수(예: $\tanh$), R_i 는 $m \times m$ 크기의 가중 요소 행렬, $\vec{b}_i$ 는 m차원 바이어스 벡터를 나타낸다. 이를 통해 입력 벡터와 출력 벡터를 행 별로 나눌 수 있으며, 각 행은 비선형 활성화 함수에 따라 자신의 고유한 선형 회귀 파라미터를 가진다.

[0066] 다음, 세로 방향에 대한 영향을 예측하기 위하여, 열 별로 독립적인 병렬 선형 레이어($PL_{col}(X)$)를 적용한다. 그 구조는 행이 아닌 열을 병렬로 처리하는 것을 제외하면 유사하다.

수학적 식 5

$$[0067] \quad PL_{col}(X) = PL_{row}(X^T)^T$$

[0068] 그리고, 두 방향에 대한 병렬 선형 레이어를 적층함으로써 행 방향과 열 방향에 모두 종속성을 가지는 형태의 수학적 식 6의 예측 모델(RCN)을 정의할 수 있다.

수학식 6

$$\text{RCN} : W_e = \text{PL}_{\text{col}}(\text{PL}_{\text{row}}(W)) \circ U$$

여기서, U 는 W_e 의 크기가 $[-1, 1]$ 의 범위에 들어오도록 하는 요소 곱 레이어(element-wise multiplication layer)를 위한 $n \times m$ 크기의 파라미터 행렬, $\circ$ 는 하다마드 곱(Hadamard product)이다. W 는 학습 데이터로 임의 생성한 가중치 행렬, W_e 는 예측 모델에 대한 출력 값인 실효 가중치 행렬을 나타낸다.

수학식 4와 5에 따른 두 레이어를 차례로 연결하고 마지막에 요소 곱 레이어를 추가하여 출력 값의 범위를 $[-1, 1]$ 로 조정하면 수학식 5와 같은 예측 모델(RCN)을 완성할 수 있다.

이에 따르면, 이진값인 0 또는 1을 요소로 갖는 임의 생성한 가중치 행렬을 입력으로 하고 SPICE 시뮬레이션을 통해 추정된 실효 가중치에 해당하는 행렬을 타겟으로 하여 실효 가중치 예측 모델(RCN)을 훈련시킬 수 있다.

또한, 실효 가중치 예측 모델(RCN)은 입력된 가중치 행렬(W)에 대응하여 실효 가중치 행렬(W_e)을 출력하도록 구성될 것을 알 수 있다.

여기서, 제1 학습부(110)는 아래 수학식 7에 의해 연산되는 평균 제곱 오차(L)를 이용하여, 실효 가중치 예측 모델에서 출력되는 실효 가중치 행렬(W_e)이 타겟 가중치 행렬($\hat{W}_e$)을 추종하도록 학습시킬 수 있다.

수학식 7

$$L = \frac{1}{N} \|\hat{W}_e - W_e\|_2^2$$

여기서, L 은 평균 제곱 오차(MSE; mean square error)에 해당하며, $N = n \times m$ 이다. 즉, 수학식 7에 따라, W_e 과 $\hat{W}_e$ 간의 평균 제곱 오차가 최소화되도록 학습될 수 있는데, 예를 들어 평균 제곱 오차가 기 설정된 기준값 이내로 들어오면 학습을 완료할 수 있다.

여기서, 타겟 가중치 행렬은 앞서 설명한 바와 같이 SPICE 시뮬레이션에 의하여 기 추정된 값에 해당한다.

이와 같이 학습이 완료된 이후부터, 실효 가중치 예측 모델은 행렬-벡터 곱셈 연산 대상이 되는 입력 벡터와 가중치 행렬이 있을 때, 해당 가중치 행렬의 입력 시에 그에 대응된 실효 가중치 행렬을 출력하여 제공할 수 있게 된다.

만일, 행렬-벡터 곱셈 연산 대상이 되는 입력 벡터와 가중치 행렬이 제1 입력 벡터와 제1 가중치 행렬이라면, 실효 가중치 예측 모델은 제1 가중치 행렬의 입력에 대응하여, 전압 강하의 영향이 반영된 실효 가중치 행렬을 정확하게 출력하여 제공할 수 있다.

여기서, 본 발명의 실시예는, 실효가중치 예측 모델의 훈련이 완료되면, 이를 원래의 타겟 뉴럴 네트워크의 훈련에 적용하여 재훈련을 진행할 수 있다. 재훈련 과정에서의 뉴럴 네트워크의 행렬곱 부분($Y=WX$)을 실효 가중치를 고려한 행렬곱($Y=h(W)X$)으로 대체함으로써 왜곡되는 정도를 훈련 과정에서 보상할 수 있다. 또한 이렇게 훈련된 네트워크를 다시 SPICE 시뮬레이션을 통해 정확도를 검증하는 과정을 거칠 수 있다.

한편, 본 발명의 실시예는 RCN 기반의 학습 모델을 활용하는 것을 대표 예시로 하고 있지만, SCN(Scaling convolutional network) 기반의 학습 모델을 사용할 수도 있다.

SCN 모델은 주변 소자의 영향을 중심으로 실효 가중치를 예측한다. 네트워크를 설계하는 과정을 단순화하기 위하여 공간적 상관 관계를 획득하고 컨볼루션 레이어를 추가하여, 메모리 셀별 오차를 보상한다. 컨볼루션 레이어는 적은 수의 파라미터로 입력의 공간 패턴을 조절하는 데 뛰어나다. 상위 레벨에서 SCN은 두 방향의 스케일

링 레이어와 그 사이에 있는 컨볼루션 네트워크로 구성된다. 내부 컨볼루션 네트워크는 n 개의 기본 블록의 스택으로 정의되고 마지막 레이어를 제외하면 각각 c 개의 출력 채널을 가진다. 마지막 레이어는 1개의 출력 채널을 가진다.

[0083] 기본 블록(CL_c)은 아래의 수식식 8과 같이 표현된다.

수식식 8

[0084] $CL_c : X \rightarrow Y$ where $Y = ReLU(Conv_c(X))$

[0085] 즉, 컨볼루션 필터 크기가 3×3 이고, 패딩(padding) 및 스트라이드(stride)가 1이며, 출력 채널이 c 개인 컨볼루션 레이어와 ReLU 활성 함수를 기본 블록(CL_c)으로 사용한다.

[0086] 그리고, 이 블록을 n 개 이어 붙이고 입력과 출력에 각각 요소곱 레이어(U_1, U_2)를 추가하면 수식식 9와 같은 SCN 모델이 구성된다.

수식식 9

[0087] $SCN_{n,c} : W_e = CL_1(CL_c(\dots(CL_c(W \circ U_1)))) \circ U_2$

[0088] 이때, 요소곱 레이어는 소자의 절대적 위치(입/출력 포트로부터의 거리)의 영향을 반영하고, 컨볼루션 레이어는 소자의 상대적 위치(주변 소자의 저항값)의 영향을 반영하여, 최종적 실패 가중치(W_e)를 예측하게 된다.

[0089] 이하에서는 RCN 모델을 이용하는 것을 대표 예시로 하여 설명한다.

[0090] 실패 가중치 예측 모델의 학습이 완료된 이후, 실패 가중치 연산부(120)는 연산 대상이 되는 제1 입력 벡터와 제1 가중치 행렬 중에서 제1 가중치 행렬을 기 학습된 실패 가중치 예측 모델에 입력시켜서 그에 대응된 실패 가중치 행렬을 도출한다. 즉, 실패 가중치 예측 모델(RCN)은 MVM 연산 대상이 되는 가중치 행렬의 입력에 대응하여 실패 가중치 행렬을 출력해 낼 수 있다.

[0091] 이후, 제2 학습부(130)는 위와 같이 도출된 실패 가중치 행렬을 타겟 뉴럴 네트워크에 적용하여 재학습하여, 훈련된 가중치 행렬을 도출한다. 훈련된 가중치 행렬은 실패 가중치 행렬로부터 뉴럴 네트워크 훈련에 의해 왜곡된 정도가 조정된 가중치 행렬을 의미한다.

[0092] 타겟 뉴럴 네트워크는 저항성 메모리(10)를 그대로 구현한 것으로 행렬-벡터 곱 연산을 수행하도록 동작하며 소프트웨어적으로 구현될 수 있다. 본 발명의 실시예에서 타겟 뉴럴 네트워크는 BinaryNet(Binarized Neural Networks)을 사용한다.

[0093] 제2 학습부(130)는 타겟 뉴럴 네트워크 상에 제2 입력 벡터를 입력 값으로 적용하고 제1 가중치 행렬에 대응된 실패 가중치 행렬을 가중치 값으로 적용할 때 도출되는 출력 벡터가 정답 값(제2 입력 벡터와 제1 가중치 행렬 간 곱셈 결과)을 추종하도록 타겟 뉴럴 네트워크의 가중치 값을 훈련시켜서, 훈련된 가중치 행렬(W')을 획득한다. 타겟 뉴럴 네트워크는 초기에는 저항성 메모리(10)와 같이 입력 벡터와 실패 가중치 간의 곱인 왜곡된 결과를 출력하며 학습이 진행될수록 정답에 가까운 값을 출력하게 된다. 이를 통해 왜곡되는 정도를 훈련 과정에서 보상한다.

[0094] 여기서, 제2 입력 벡터는 타겟 뉴럴 네트워크의 학습에 사용되는 학습용 데이터인데, 이는 실제 추론에 사용된 제1 입력 벡터와는 상이하다. 이와 같이 실제 추론에 사용되는 제1 입력 벡터와 타겟 뉴럴 네트워크의 학습에 사용되는 제2 입력 벡터를 달리 적용하게 되면 다양한 입력 패턴에 강인한 추론 성능을 얻게 된다.

[0095] 제어부(140)는 행렬-곱 연산 대상인 제1 입력 벡터와 제1 가중치 행렬(W) 대신, 제1 입력 벡터와 훈련된 가중치 행렬(W')을 저항성 메모리(10)에 매핑하여 행렬-벡터 곱 연산을 수행한다. 즉, 제어부(140)는 제1 입력 벡터를 저항성 메모리(10)의 비트 라인에 적용하고 훈련된 가중치 행렬을 저항성 메모리(10) 내 저항성 소자의 컨덕턴스 값으로 적용한다.

- [0096] 이에 따라, 저항성 메모리(10)에서는 제1 입력 벡터와 훈련된 가중치 행렬을 곱한 결과를 출력 값으로 도출하여 제공한다. 이러한 훈련된 가중치는 IR 드롭(전압 강하)에 의한 출력 왜곡 문제를 해소한다. 이와 같이 훈련에 의해 새롭게 구해진 가중치 행렬(W')을 원래 사용할 가중치 행렬(W) 대신 저항성 메모리(10) 내 저항성 소자의 각 컨덕턴스 값으로 적용할 경우에 전압 드롭의 영향이 상쇄된 정확한 출력값 Y 또는 그와 매우 근접한 값을 얻을 수 있다.
- [0097] 이와 같은, 본 발명에 따르면, 저항성 메모리 기반의 뉴럴 네트워크 가속기에서 입력 벡터와 가중치 행렬 간의 행렬 곱 연산 시에 저항성 메모리의 선저항으로 인한 전압 강하의 왜곡이 반영된 실효 가중치 행렬을 예측하고 이를 타겟 뉴럴 네트워크에 적용하고 재학습함으로써 왜곡을 보정할 수 있는 새로운 가중치 행렬을 도출한다.. 이를 통해 저항성 메모리 기반의 뉴럴 네트워크 가속기에 대한 추론 성능을 향상시킬 수 있다.
- [0098] 도 4는 도 2를 이용한 연산 오차 보정 방법을 설명하는 도면이다.
- [0099] 먼저, 제1 학습부(110)는 훈련 데이터와 그에 대응된 타겟 가중치 행렬을 이용하여, 임의 가중치 행렬의 입력에 대응하여 실효 가중치 행렬을 출력하는 실효 가중치 예측 모델을 학습시킨다(S410).
- [0100] 구체적으로, 제1 학습부(110)는 임의 가중치 행렬을 입력 값으로 하고 그에 대응하는 실효 가중치 행렬을 출력 값으로 하는 실효 가중치 예측 모델을 구성하고 해당 예측 모델을 학습시킨다. 이때, 실효 가중치 예측 모델의 출력 값인 실효 가중치 행렬이 해당 학습 데이터에 매칭된 타겟 가중치 행렬을 추종하도록 예측 모델을 학습시킨다.
- [0101] 학습이 완료된 이후에, 실효 가중치 연산부(120)는 연산 대상이 되는 제1 입력 벡터와 제1 가중치 행렬 중에서 제1 가중치 행렬을 실효 가중치 예측 모델에 입력시켜서 해당 예측 모델로부터 실효 가중치 행렬을 획득한다(S420).
- [0102] S410 단계에 의해 사전 학습된 실효 가중치 예측 모델은 제1 가중치 행렬에 대해 저항성 메모리의 전압 강하가 반영된 상태의 실효 가중치 행렬을 출력 값으로 제공한다.
- [0103] 이후, 제2 학습부(130)는 제2 입력 벡터와 S420 단계에서 획득된 실효 가중치 행렬을 타겟 뉴럴 네트워크에 적용하여 훈련한다(S430). 이에 따라 가중치 값이 조정되면서 훈련에 의한 새로운 가중치 행렬을 얻게 된다. 훈련 과정에서 매 가중치 업데이트 마다 S430 ~ S440 이 반복될 수 있다.
- [0104] 이후, 제어부(140)는 제1 입력 벡터와 훈련된 새로운 가중치 행렬을 저항성 메모리(10)에 매핑하여 MVM 연산을 수행한다(S440). 예를 들어, 제어부(140)는 저항성 메모리(10)에 인가되는 신호를 제어할 수 있으며, 이를 통해 저항성 메모리(10)의 비트 라인(L1)에 제1 입력 벡터를, 저항성 소자(R)에 훈련된 가중치 행렬을 매핑할 수 있다.
- [0105] 이에 따라 저항성 메모리(10)는 제1 입력 벡터와 훈련된 가중치 행렬을 곱셈 연산함으로써 선저항으로 인한 연산 왜곡이 보정된 출력 벡터를 도출할 수 있게 한다. 물론, 이에 따르면, 제1 가중치 행렬을 그대로 사용하는 경우 발생하는 연산 왜곡 문제를 해결할 수 있다.
- [0106] 도 5는 본 발명의 실시예에 따른 예측 모델의 정확도를 시뮬레이션을 통해 검증한 결과를 나타낸다.
- [0107] 이러한 도 5는 SPICE 시뮬레이션을 통해 검증한 정확도(validation accuracy) 결과이다. 이때, 다양한 크기(32×32 , 64×64)의 저항성 메모리 크로스바 어레이(RCA)에 대한 성능 실험을 수행하였다.
- [0108] 이때, 타겟 뉴럴 네트워크는 BinaryNet(Binarized Neural Networks)이며, 네트워크에서 사용한 데이터셋은 MNIST(Modified National Institute of Standards and Technology database), CIFAR10(Canadian Institute For Advanced Research 10), SVHN(Street View House Numbers)으로 다양하다.
- [0109] 도 5에서 w/o ret은 별도의 왜곡 보정을 수행하지 않은 경우이고, Mask 기법과 NIA 기법은 종래의 실효 가중치 예측 기법이며, RCN과 SCN은 본 발명에 따른 실효 가중치 예측 기법에 해당한다. 도 5의 결과로부터, RCN과 SCN 예측 모델을 사용할 경우 기존(Mask 방법, NIA 방법)에 비해 월등한 정확도 상승을 달성한 것을 확인할 수 있다. 그래프의 점선은 선저항이 없을 때의 이상적인 정확도 값(baseline accuracy)을 나타내는데, RCN과 SCN은 선저항이 높은 조건 하에서도 매우 뛰어난 성능을 보인다.
- [0110] 이상과 같은 본 발명에 따르면, 저항성 메모리 기반의 뉴럴 네트워크 가속기에서 입력 벡터와 가중치 행렬 간의 행렬 곱 연산 시에 저항성 메모리의 선저항으로 인한 행렬곱 계산 결과의 왜곡을 보정함으로써 저항성 메모리

기반의 뉴럴 네트워크 가속기에 대한 추론 성능을 향상시킬 수 있다.

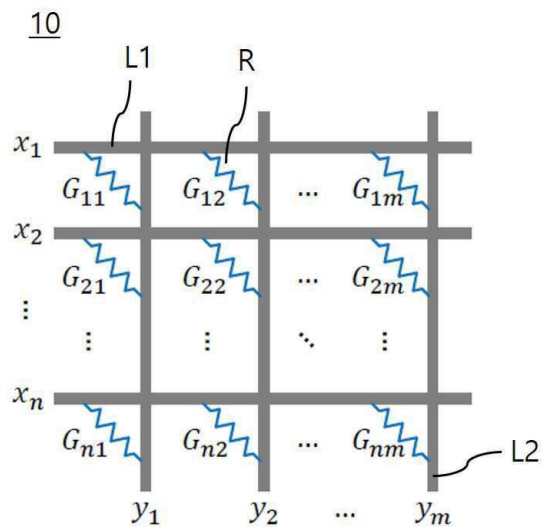
[0111] 본 발명은 도면에 도시된 실시 예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 다른 실시 예가 가능하다는 점을 이해할 것이다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 특허청구범위의 기술적 사상에 의하여 정해져야 할 것이다.

부호의 설명

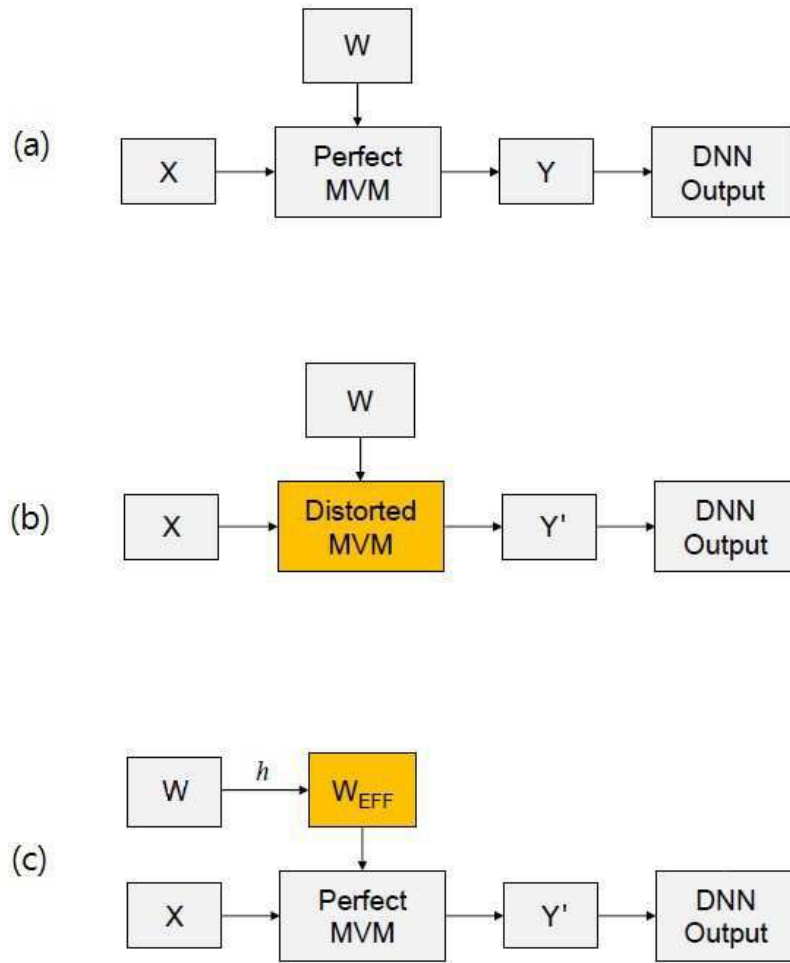
[0112] 100: 연산 오차 보정 장치 110: 제1 학습부
120: 실패 가중치 연산부 130: 제2 학습부
140: 제어부

도면

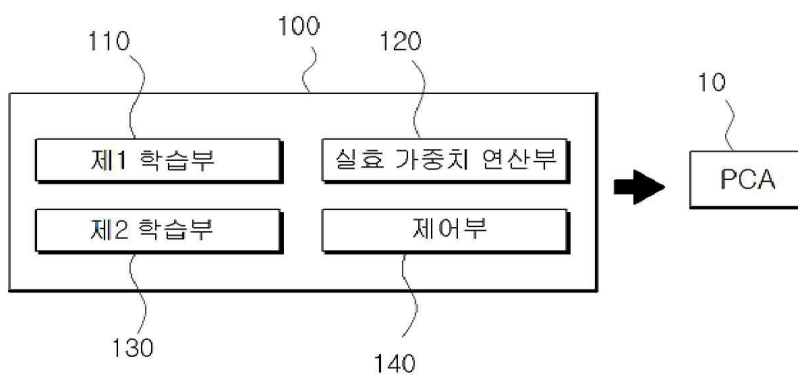
도면1



도면2



도면3



도면4

