(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2023년09월25일
(11) 등록번호 10-2582079
(24) 등록일자 2023년09월19일

(51) 국제특허분류(Int. Cl.)
G06N 3/08 (2023.01) *G06N 3/04* (2023.01)
(52) CPC특허분류
G06N 3/08 (2023.01)
G06N 3/04 (2023.01)
(21) 출원번호 10-2020-0102311
(22) 출원일자 2020년08월14일
심사청구일자 2020년08월14일
(65) 공개번호 10-2022-0002020
(43) 공개일자 2022년01월06일
(30) 우선권주장
1020200079782 2020년06월30일 대한민국(KR)
(56) 선행기술조사문헌
KR1020190128980 A*
US20190164538 A1*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
세종대학교산학협력단
서울특별시 광진구 능동로 209 (군자동, 세종대학교)
(72) 발명자
박기호
서울특별시 노원구 중계로 184, 101동 903호(중계동, 라이프청구신동아아파트)
한치원
서울특별시 중구 다산로14길 27-4(신당동)
기민관
서울특별시 동대문구 서울시립대로 14, 104동 1104호(답십리동, 청계한신휴플러스)
(74) 대리인
민영준

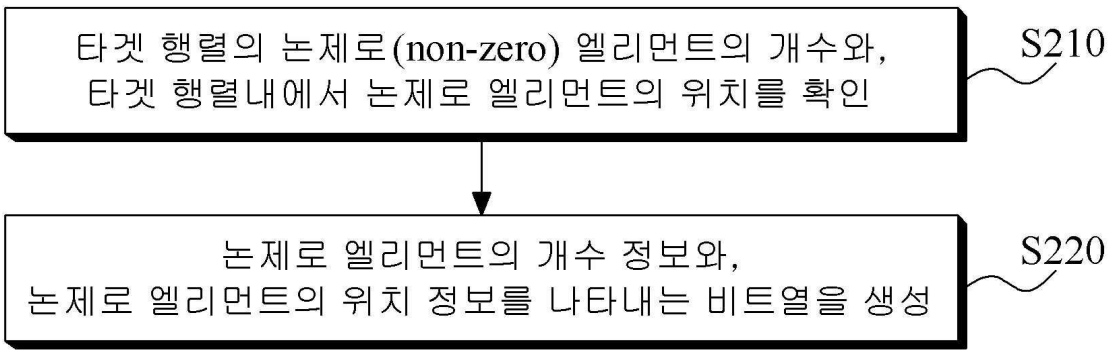
전체 청구항 수 : 총 13 항 심사관 : 이준상

(54) 발명의 명칭 **행렬 인덱스 정보 생성 방법, 행렬 인덱스 정보를 이용하는 행렬 처리 방법, 장치**

(57) 요 약

희소 행렬을 포함한 타겟 행렬에 대한 행렬 인덱스 정보를 생성하는 방법과 행렬 인덱스 정보를 이용하는 행렬을 처리하는 방법이 개시된다. 개시된 행렬 인덱스 정보 생성 방법은, 타겟 행렬의 엘리먼트를 확인하는 단계; 및 상기 엘리먼트 각각에 할당되며, 상기 타겟 행렬 내에서의 상기 엘리먼트의 위치 정보를 나타내는 적어도 하나의 비트를 포함하는 비트열을 생성하는 단계를 포함한다.

대 표 도 - 도2



이 발명을 지원한 국가연구개발사업

과제고유번호	1711115123
과제번호	2018R1A2B6002534
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	개인기초연구(과기정통부)(R&D)
연구과제명	차세대 응용을 위한 데이터 중심 가속기 기반 컴퓨팅 시스템 구조 설계
기 여 율	1/1
과제수행기관명	세종대학교 산학협력단
연구기간	2020.03.01 ~ 2021.02.28

명 세 서

청구범위

청구항 1

행렬 처리 장치에 의해 수행되는, 행렬 인덱스 정보 생성 방법에 있어서,

타겟 행렬의 엘리먼트를 확인하는 단계; 및

상기 확인된 엘리먼트 각각에 할당되며, 상기 타겟 행렬 내에서의 상기 엘리먼트의 위치 정보를 나타내는 적어도 하나의 비트를 포함하는 비트열을 생성하는 단계를 포함하며,

상기 비트열은

상기 엘리먼트 중 논제로(non-zero) 엘리먼트의 개수 정보를 나타내는 제1비트열; 및

상기 위치 정보를 나타내는 제2비트열을 포함하며,

상기 제1비트열은, 상기 논제로 엘리먼트의 개수가 이진수로 표현된 비트열이며,

상기 제2비트열의 비트 각각은, 상기 타겟 행렬내에서의 엘리먼트의 위치 각각에 대응되는

행렬 인덱스 정보 생성 방법.

청구항 2

삭제

청구항 3

제 1항에 있어서,

상기 제2비트열은,

상기 타겟 행렬내에서의 엘리먼트의 위치 각각에 대응되는 비트를 포함하며,

상기 제2비트열에서, 상기 타겟 행렬의 제로 엘리먼트의 위치에 대응되는 비트값과, 상기 논제로 엘리먼트의 위치에 대응되는 비트값은, 서로 상이한

행렬 인덱스 정보 생성 방법.

청구항 4

제 1항에 있어서,

상기 타겟 행렬은

인공 신경망의 가중치값을 포함하는 가중치 행렬인

행렬 인덱스 정보 생성 방법.

청구항 5

삭제

청구항 6

제1타겟 행렬에 대한 행렬 인덱스 정보를 이용하여, 상기 제1타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩

하고, 상기 행렬 인덱스 정보를 이용하여, 제2타겟 행렬의 엘리먼트 중에서, 상기 제1타겟 행렬의 논제로 엘리먼트와의 곱셈 대상인 엘리먼트 값을 상기 메모리에서 로딩하는 단계; 및

상기 로딩된 논제로 엘리먼트값 및 상기 곱셈 대상인 엘리먼트 값을 연산기로 전달하는 단계를 포함하며,

상기 행렬 인덱스 정보는

상기 제1타겟 행렬의 논제로 엘리먼트에 대한 개수 정보와, 상기 제1타겟 행렬내에서 상기 논제로 엘리먼트의 위치 정보를 포함하는,

행렬 인덱스 정보를 이용하는 행렬 처리 방법.

청구항 7

제 6항에 있어서,

상기 제1타겟 행렬은 인공 신경망의 가중치값을 포함하는 행렬이며,

상기 제2타겟 행렬은 상기 가중치값의 활성화 여부를 결정하는 엘리먼트를 포함하는 행렬인

행렬 인덱스 정보를 이용하는 행렬 처리 방법.

청구항 8

타겟 행렬에 대한 행렬 인덱스 정보를 이용하여, 상기 타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩하는 단계; 및

상기 로딩된 논제로 엘리먼트값을 연산기로 전달하는 단계를 포함하며,

상기 행렬 인덱스 정보는

상기 타겟 행렬의 논제로 엘리먼트에 대한 개수 정보와, 상기 타겟 행렬내에서 상기 논제로 엘리먼트의 위치 정보를 포함하며,

상기 논제로 엘리먼트값을 메모리에서 로딩하는 단계는

상기 행렬 인덱스 정보를 이용하여, 상기 타겟 행렬의 논제로 엘리먼트값에 대한 주소값을 결정하고, 상기 주소값을 이용하여, 상기 타겟 행렬의 논제로 엘리먼트값을 로딩하는

행렬 인덱스 정보를 이용하는 행렬 처리 방법.

청구항 9

제 8항에 있어서,

상기 논제로 엘리먼트값에 할당된 메모리 주소값은, 미리 설정된 규칙에 따라 연속된 형태이며,

상기 논제로 엘리먼트값을 메모리에서 로딩하는 단계는

상기 타겟 행렬의 논제로 엘리먼트값보다 이전에 상기 메모리에서 로딩된 논제로 엘리먼트값의 개수를 이용하여, 상기 타겟 행렬의 논제로 엘리먼트값에 대한 주소값을 결정하는 단계; 및

상기 주소값을 이용하여, 상기 타겟 행렬의 논제로 엘리먼트값을 로딩하는 단계

를 포함하는 행렬 인덱스 정보를 이용하는 행렬 처리 방법.

청구항 10

제1타겟 행렬에 대한 행렬 인덱스 정보를 이용하여, 상기 제1타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩

하는 단계; 및

상기 로딩된 논제로 엘리먼트값을 연산기로 전달하는 단계를 포함하며,

상기 행렬 인덱스 정보는

상기 제1타겟 행렬의 논제로 엘리먼트에 대한 개수 정보와, 상기 제1타겟 행렬내에서 상기 논제로 엘리먼트의 위치 정보를 포함하며,

상기 논제로 엘리먼트값을 메모리에서 로딩하는 단계는

제2타겟 행렬에 대한 행렬 인덱스 정보를 이용하여, 상기 제2타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩하며,

상기 로딩된 논제로 엘리먼트값을 연산기로 전달하는 단계는

상기 제1 및 제2타겟 행렬에 대한 행렬 인덱스 정보를 상기 연산기로 전달하는

행렬 인덱스 정보를 이용하는 행렬 처리 방법.

청구항 11

타겟 행렬에 대한 행렬 인덱스 정보를 이용하여, 상기 타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩하는 단계; 및

상기 로딩된 논제로 엘리먼트값을 연산기로 전달하는 단계를 포함하며,

상기 행렬 인덱스 정보는

상기 타겟 행렬의 논제로 엘리먼트에 대한 개수 정보와, 상기 타겟 행렬내에서 상기 논제로 엘리먼트의 위치 정보를 포함하며,

상기 로딩된 논제로 엘리먼트값을 연산기로 전달하는 단계는

상기 행렬 인덱스 정보 및 상기 논제로 엘리먼트값을 이용하여, 상기 타겟 행렬에 제로값을 패딩하여 상기 타겟 행렬을 복원하는 단계; 및

상기 복원된 타겟 행렬을 상기 연산기로 전달하는 단계

를 포함하는 행렬 인덱스 정보를 이용하는 행렬 처리 방법.

청구항 12

타겟 행렬에 대한 행렬 인덱스 정보를 이용하여, 상기 타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩하는 단계; 및

상기 로딩된 논제로 엘리먼트값을 연산기로 전달하는 단계를 포함하며,

상기 행렬 인덱스 정보는

상기 타겟 행렬의 논제로 엘리먼트에 대한 개수 정보와, 상기 타겟 행렬내에서 상기 논제로 엘리먼트의 위치 정보를 포함하며,

상기 로딩된 논제로 엘리먼트값을 연산기로 전달하는 단계는

상기 연산기의 개수와, 상기 논제로 엘리먼트의 개수를 비교하는 단계; 및

상기 비교 결과에 따라서, 상기 로딩된 논제로 엘리먼트값을 연산기로 전달하는 단계

를 포함하는 행렬 인덱스 정보를 이용하는 행렬 처리 방법.

청구항 13

제 12항에 있어서,

상기 로딩된 논제로 엘리먼트값을 연산기로 전달하는 단계는

상기 로딩된 논제로 엘리먼트값의 개수가 상기 연산기의 개수보다 적은 경우, 상기 타겟 행렬의 논제로 엘리먼트값 이후에 상기 메모리에서 로딩되는 논제로 엘리먼트값을, 상기 타겟 행렬의 논제로 엘리먼트값과 함께 상기 연산기로 전달하는

행렬 인덱스 정보를 이용하는 행렬 처리 방법.

청구항 14

타겟 행렬의 엘리먼트 각각에 할당되며, 상기 타겟 행렬 내에서의 상기 엘리먼트의 위치 정보를 나타내는 비트를 포함하는 적어도 하나의 비트열을 생성하는 비트열 생성부;

상기 비트열을 이용하여, 상기 엘리먼트 중 논제로 엘리먼트의 값을 메모리에서 로딩하는 데이터 로딩부; 및

상기 로딩된 데이터를 이용하여, 상기 타겟 행렬에 대한 연산을 수행하는 연산부를 포함하며,

상기 비트열은

상기 엘리먼트 중 논제로(non-zero) 엘리먼트의 개수 정보를 나타내는 제1비트열; 및

상기 위치 정보를 나타내는 제2비트열을 포함하며,

상기 제1비트열은, 상기 논제로 엘리먼트의 개수가 이진수로 표현된 비트열이며,

상기 제2비트열의 비트 각각은, 상기 타겟 행렬내에서의 엘리먼트의 위치 각각에 대응되는

행렬 인덱스 정보를 이용하는 행렬 처리 장치.

청구항 15

삭제

청구항 16

제 14항에 있어서,

상기 메모리는

상기 타겟 행렬에 대한 비트열 및 상기 논제로 엘리먼트값을 저장하는

행렬 인덱스 정보를 이용하는 행렬 처리 장치.

발명의 설명

기 술 분 야

[0001] 본 발명은 행렬의 인덱스 정보를 생성하는 방법과, 행렬의 인덱스 정보를 이용하여 행렬을 처리하는 방법 및 장치에 관한 것이다.

배 경 기 술

[0003] 최근 이미지 인식 등의 서비스 분야에서 활용되는 CNN(Convolutional Neural Network) 모델과 같은 신경망 모델이 발전함에 따라서, 신경망 모델이 처리해야 하는 레이어의 깊이 등이 증가하고 있다. 이러한 요인들로 인하여 신경망 모델의 가중치 행렬과 같은 파라미터의 수가 증가하게 되어, 높은 메모리 오버헤드가 중요한 이슈로 대

두되었다.

[0004] 이를 해결하기 위한 방안으로, 신경망 모델의 과적합 문제를 해결하기 위해 수행하는 프루닝(pruning) 기법이, 가중치 행렬을 희소 행렬(sparse matrix)로 만든다는 것을 활용하여, 희소 행렬에 대한 연산을 효율적으로 수행할 수 있는 행렬의 인덱싱 방법에 대한 연구들이 진행되었다.

[0005] 희소 행렬에 대한 인덱싱 방법으로 CSR(Compressed Sparse Row)이 많이 활용되고 있는데, CSR과 같은 희소 행렬 인덱싱 방법은 가중치 행렬 단위로 적용하였을 때 인덱스 크기 및 위치 확인을 위한 연산이 필요하다는 점과, 희소성이 낮은 즉, 논제로(non-zero) 엘리먼트의 개수가 적은 행렬의 표현에는 상당한 오버헤드가 발생한다는 단점이 있다.

[0006] 관련 선행문헌으로 대한민국 공개특허 제2020-0052182호, 제2018-0067426호가 있다.

발명의 내용

해결하려는 과제

[0008] 본 발명은 희소 행렬을 포함한 타겟 행렬에 대한 행렬 인덱스 정보를 생성하는 방법을 제공하기 위한 것이다.

[0009] 또한 본 발명은 타겟 행렬에 대한 행렬 인덱스 정보를 이용하여 메모리로부터 타겟 행렬에 대한 정보를 로딩하고 행렬을 처리하는 방법 및 장치를 제공하기 위한 것이다.

과제의 해결 수단

[0011] 상기한 목적을 달성하기 위한 본 발명의 일 실시예에 따르면, 타겟 행렬의 엘리먼트를 확인하는 단계; 및 상기 엘리먼트 각각에 할당되며, 상기 타겟 행렬 내에서의 상기 엘리먼트의 위치 정보를 나타내는 적어도 하나의 비트를 포함하는 비트열을 생성하는 단계를 포함하는 행렬 인덱스 정보 생성 방법이 제공된다.

[0012] 또한 상기한 목적을 달성하기 위한 본 발명의 다른 실시예에 따르면, 제1타겟 행렬에 대한 행렬 인덱스 정보를 이용하여, 상기 제1타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩하는 단계; 및 상기 로딩된 데이터를 연산기로 전달하는 단계를 포함하며, 상기 행렬 인덱스 정보는 상기 제1타겟 행렬의 논제로 엘리먼트에 대한 개수 정보와, 상기 제1타겟 행렬내에서 상기 논제로 엘리먼트의 위치 정보를 포함하는 행렬 인덱스 정보를 이용하는 행렬 처리 방법이 제공된다.

[0013] 또한 상기한 목적을 달성하기 위한 본 발명의 또 다른 실시예에 따르면, 타겟 행렬의 엘리먼트 각각에 할당되며, 상기 타겟 행렬 내에서의 상기 엘리먼트의 위치 정보를 나타내는 적어도 하나의 비트를 포함하는 비트열을 생성하는 비트열 생성부; 상기 비트열을 이용하여, 상기 엘리먼트 중 논제로 엘리먼트의 값을 메모리에서 로딩하는 데이터 로딩부; 및 상기 로딩된 데이터를 이용하여, 상기 타겟 행렬에 대한 연산을 수행하는 연산부를 포함하는 행렬 인덱스 정보를 이용하는 행렬 처리 장치가 제공된다.

발명의 효과

[0015] 본 발명의 일실시예에 따르면, 행렬의 희소성이 감소하더라도 행렬 인덱스 정보의 크기가 일정하게 유지될 수 있으므로, 메모리 사용량을 줄일 수 있다.

[0016] 또한 본 발명의 일실시예에 따르면, 타겟 행렬의 전체 엘리먼트의 개수 정보와 위치 정보가 행렬 인덱스 정보에 포함되어 있으므로, 행렬 인덱스 정보에 대한 한번의 메모리 접근으로, 타겟 행렬의 정보를 획득할 수 있으며, 따라서 타겟 행렬의 정보 획득을 위한 메모리 접근 횟수가 줄어들 수 있다.

도면의 간단한 설명

[0018] 도 1은 행렬 인덱싱 방법 중 하나인 CSR을 설명하기 위한 도면이다.

도 2는 본 발명의 일실시예에 따른 행렬 인덱스 정보 생성 방법을 설명하기 위한 도면이다.

도 3은 본 발명의 일실시예에 따른 행렬 인덱스 정보를 나타내는 도면이다.

도 4는 본 발명의 일실시예에 따른 행렬 인덱스 정보의 크기를 설명하기 위한 도면이다.

도 5는 본 발명의 일실시예에 따른 행렬 인덱스 정보를 이용하는 행렬 처리 장치를 설명하기 위한 도면이다.

도 6은 본 발명의 일실시예에 따른 행렬 인덱스 정보를 이용하는 행렬 처리 방법을 설명하기 위한 도면이다.

도 7은 메모리에 저장된 행렬 인덱스 정보의 일예를 나타내는 도면이다.

도 8은 본 발명의 다른 실시예에 따른 행렬 인덱스 정보를 이용하는 행렬 처리 방법을 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0019] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.

[0020] 이하에서, 본 발명에 따른 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다.

[0022] 도 1은 행렬 인덱싱 방법 중 하나인 CSR을 설명하기 위한 도면이다.

[0023] CSR에 따르면, 행렬의 행(row) 단위로 인덱싱이 이루어진다. 도 1과 같이, 논제로 엘리먼트 a,b,c,d 이외 0을 포함하는 3x3 크기의 타겟 행렬(100)이 주어진 경우, CSR에 따르면, 3개의 행 각각에 대한 인덱싱이 이루어져, 행 및 열에 대한 인덱스 정보가 생성된다. 행에 대한 인덱스 정보에는, 행 각각에 대한 논제로 엘리먼트(non-zero element)의 개수의 누적 정보가 포함되며, 열에 대한 인덱스 정보에는 각 행에서 논제로 엘리먼트의 위치 정보가 포함된다.

[0024] 제1행(110)에서 논제로 엘리먼트(a)는 1개이며, 제2행(120)에서 논제로 엘리먼트(b, c)는 2개이다. 그리고 제3행(130)에서 논제로 엘리먼트(d)는 1개이다. 따라서, 행에 대한 인덱스 정보(140)는, 제1행(110)의 논제로 엘리먼트 개수에 대응되는 인덱스 1, 제1행(110)의 논제로 엘리먼트 개수에 제2행(120)의 논제로 엘리먼트의 개수가 누적된 값에 대응되는 인덱스 3, 제1 및 제2행(110, 120)의 논제로 엘리먼트의 누적 개수에, 제3행(130)의 논제로 엘리먼트의 개수가 더해진 값에 대응되는 인덱스 4를 포함한다.

[0025] 그리고 제1행(110)에서, 논제로 엘리먼트(a)는 제1열에 위치하며, 제2행(120)에서 논제로 엘리먼트(b, c)는 제2 및 제3열에 위치한다. 마지막으로 제3행(130)에서 논제로 엘리먼트(d)는 제3열에 위치한다. 따라서, 열에 대한 인덱스 정보(150)는 제1행(110)에서 제1열의 위치에 대응되는 인덱스 0, 제2행(120)에서 제2 및 제3열의 위치에 대응되는 인덱스 1 및 2, 마지막으로 제3행(130)에서 제3열의 위치에 대응되는 인덱스 2를 포함한다.

[0026] CSR은 희소성이 매우 높은 행렬을 타겟팅하여 만들어진 행렬 인덱싱 방법이기 때문에, 타겟 행렬의 희소성이 적을 경우, 즉 타겟 행렬에서 논제로 엘리먼트의 개수가 많을 경우, 행렬 인덱스 정보의 크기가 증가하는 문제가 있다. 또한 CSR의 경우, 행렬 인덱스 정보를 이용하여 타겟 행렬의 정보를 획득하기 위해서는, 타겟 행렬의 행의 개수만큼 메모리 접근이 필요하다.

[0027] 이에 본 발명은, 타겟 행렬의 희소성이 낮아지더라도 크기가 일정하게 유지될 수 있으며, 타겟 행렬의 정보를 획득하기 위한 메모리 접근 횟수를 줄일 수 있는 행렬 인덱스 정보 생성 방법을 제안한다. 그리고 이와 함께 행렬 인덱스 정보를 이용하는 행렬 처리 방법을 제안한다.

[0028] 본 발명의 일실시예는, 타겟 행렬의 엘리먼트를 확인하고, 엘리먼트 각각에 할당되며, 타겟 행렬 내에서의 엘리먼트의 위치 정보를 나타내는 적어도 하나의 비트를 포함하는 비트열 즉, 행렬 인덱스 정보를 생성한다. 즉, 본 발명의 일실시예는 타겟 행렬의 엘리먼트 각각에 할당되어 엘리먼트 각각에 대응되는 비트로 이루어진 비트열을 생성하며, 비트열의 각 비트는 타겟 행렬 내에서의 엘리먼트의 위치를 나타낸다.

[0029] 이러한 행렬 인덱스 정보는, 실시예에 따라서, 타겟 행렬의 엘리먼트 중에서 논제로 엘리먼트의 개수 정보를 나타내는 비트열과, 타겟 행렬의 모든 엘리먼트에 대한 위치 정보를 나타내는 비트열을 포함할 수 있다.

[0030] 본 발명의 일실시예에 따른 행렬 인덱스 정보 생성 방법과, 행렬 인덱스 정보를 이용하는 행렬 처리 방법은 행렬 처리 장치에서 수행될 수 있다. 이러한 행렬 처리 장치는, 프로세서나 딥러닝 가속기 등과 같은 연산용 반도체 칩이거나 또는 이러한 연산용 반도체 칩을 포함하는 컴퓨팅 장치일 수 있다.

[0032] 도 2는 본 발명의 일실시예에 따른 행렬 인덱스 정보 생성 방법을 설명하기 위한 도면이며, 도 3은 본 발명의 일실시예에 따른 행렬 인덱스 정보를 나타내는 도면이다.

[0033] 도 2를 참조하면, 본 발명의 일실시예에 따른 행렬 처리 장치는 타겟 행렬의 논제로(non-zero) 엘리먼트의 개수와, 타겟 행렬내에서 논제로 엘리먼트의 위치를 확인(S210)하고, 논제로 엘리먼트의 개수 정보와, 논제로 엘리

먼트의 위치 정보를 나타내는 비트열, 즉 행렬 인덱스 정보를 생성(S220)한다. 일실시예로서, 타겟 행렬은 인공 신경망의 가중치값을 포함하는 가중치 행렬일 수 있다.

[0034] 도 3에 도시된 바와 같이, 본 발명의 일실시예에 따른 행렬 인덱스 정보(350)는 비트열 형태로 표현되며, 논제로 엘리먼트의 개수 정보를 나타내는 제1비트열(351)과, 논제로 엘리먼트의 위치 정보를 나타내는 제2비트열(352)을 포함할 수 있다.

[0035] 도 3과 같이, 3x3 크기이며 논제로 엘리먼트 a, b, c 이외 0을 포함하는 타겟 행렬(310)이 주어진 경우, 논제로 엘리먼트(a, b, c)의 개수는 3이므로, 제1비트열(351)의 비트값은 3에 대응되는 '0011'이 된다.

[0036] 제2비트열(352)은 타겟 행렬내에서의 엘리먼트의 위치 각각에 대응되는 비트를 포함한다. 즉, 제2비트열(352)의 각 비트는, 타겟 행렬(310) 내에서의 엘리먼트 각각의 위치에 대응된다. 도 3과 같은 예시에서, 타겟 행렬(310)의 제1행, 제1열에 배치된 논제로 엘리먼트 a의 위치에 대응되는 비트는, 제2비트열(352)의 최상위 비트이며, 타겟 행렬(310)의 제2행, 제2열에 배치된 논제로 엘리먼트 b의 위치에 대응되는 비트는 제2비트열(352)의 중간에 위치한 비트이다. 그리고 타겟 행렬(310)의 제3행, 제3열에 배치된 논제로 엘리먼트 c의 위치에 대응되는 비트는 제2비트열(352)의 최하위 비트이다.

[0037] 제2비트열(352)에 포함된 비트의 개수는, 타겟 행렬 내에서의 엘리먼트의 개수 이상일 수 있으며, 도 3의 예시에서는, 타겟 행렬의 엘리먼트의 개수가 9이므로, 제2비트열(352)에 9개의 비트가 사용되었다.

[0038] 그리고 제2비트열(352)에서, 타겟 행렬(310)의 제로 엘리먼트의 위치에 대응되는 비트값과, 논제로 엘리먼트의 위치에 대응되는 비트값은, 서로 상이하게 할당된다. 따라서, 제2비트열(352)의 비트값을 확인하면, 타겟 행렬(310)의 어느 엘리먼트가 논제로 엘리먼트인지 확인할 수 있다. 도 3에 도시된 바와 같이, 제로 엘리먼트의 위치에 대응되는 비트값으로 0이 할당되고, 논제로 엘리먼트의 위치에 대응되는 비트값으로 1이 할당될 수 있다.

[0040] 도 4는 본 발명의 일실시예에 따른 행렬 인덱스 정보의 크기를 설명하기 위한 도면으로서, 논제로 엘리먼트의 개수에 따른 크기를, CSR 방법에 따라 생성된 행렬 인덱스 정보의 크기와 비교한 그래프이다.

[0041] 도 4(a)는 3x3 행렬에서의 행렬 인덱스 정보의 크기를 비교하는 그래프이며, 도 4(b)s는 7x7 행렬에서의 행렬 인덱스 정보의 크기를 비교하는 그래프이다. 도 4에서, X축은 논제로 엘리먼트의 개수를 나타내며, Y축은 행렬 인덱스 정보의 크기를 나타낸다.

[0042] 도 4에 도시된 바와 같이, 본 발명의 일실시예(non-zero bitmap indexing)에 따른 행렬 인덱스 정보의 크기는, 논제로 엘리먼트의 개수가 증가하더라도 일정하게 유지되는 반면, CSR 방법에 따른 행렬 인덱스 정보의 크기는 선형적으로 증가함을 알 수 있다.

[0043] 결국, 본 발명의 일실시예에 따르면, 행렬의 희소성이 감소하더라도 행렬 인덱스 정보의 크기가 일정하게 유지될 수 있으므로, 메모리 사용량을 줄일 수 있다.

[0044] 특히, 인공 신경망에 대한 프루닝 비율에 따라서, 가중치 행렬의 희소성은 달리지며, 프루닝 비율이 낮아질수록 가중치 행렬의 희소성은 감소하는 패턴을 나타내며, 프루닝된 모델의 가중치 행렬 별로 희소성 패턴은 큰 차이를 보일 수 있는데, 이러한 환경에서도 본 발명의 일실시예는, 일정한 크기의 행렬 인덱스 정보를 제공할 수 있으므로, 메모리 사용량을 줄일 수 있다.

[0045] 또한 본 발명의 일실시예에 따르면, 타겟 행렬의 전체 엘리먼트의 개수 정보와 위치 정보가 행렬 인덱스 정보에 포함되어 있으므로, 행렬 인덱스 정보에 대한 한번의 메모리 접근으로, 타겟 행렬의 정보를 획득할 수 있으며, 따라서 타겟 행렬의 정보 획득을 위한 메모리 접근 횟수가 줄어들 수 있다.

[0047] 도 5는 본 발명의 일실시예에 따른 행렬 인덱스 정보를 이용하는 행렬 처리 장치를 설명하기 위한 도면이다.

[0048] 도 5를 참조하면, 본 발명의 일실시예에 따른 행렬 처리 장치는 비트열 생성부(510), 데이터 로딩부(520) 및 연산부(530)를 포함한다. 실시예에 따라서 메모리를 더 포함할 수 있다.

[0049] 비트열 생성부(510)는 제1타겟 행렬의 논제로 엘리먼트의 개수 정보와, 논제로 엘리먼트의 위치 정보를 나타내는 비트열을 생성한다. 비트열은 전술된 실시예의 행렬 인덱스 정보에 대응되며, 생성된 비트열과, 타겟 행렬의 논제로 엘리먼트값은 제1메모리(540)에 저장될 수 있다.

[0050] 데이터 로딩부(520)는 비트열을 이용하여, 제1타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩한다. 데이터 로딩부(520)는 메모리에 저장된 논제로 엘리먼트값에 대한 메모리 주소값을 이용하여, 제1타겟 행렬의 논제로

엘리먼트값을 로딩할 수 있다.

[0051] 일실시예로서, 논제로 엘리먼트값에 할당된 메모리 주소값은, 미리 설정된 규칙에 따라 연속된 형태일 수 있으며, 타겟 행렬에 할당된 인덱스의 순서에 대응되도록, 복수의 타겟 행렬의 논제로 엘리먼트값에 대한 메모리 주소값은 연속적인 패턴으로 할당될 수 있다. 따라서, 데이터 로딩부(520)는 이전에 메모리에서 로딩된 논제로 엘리먼트값의 개수를 이용하여 제1타겟 행렬의 논제로 엘리먼트값의 주소값을 결정할 수 있으며, 결정된 메모리 주소값을 이용하여, 메모리로부터 제1타겟 행렬의 논제로 엘리먼트값을 로딩할 수 있다.

[0052] 연산부(530)는 로딩된 데이터를 이용하여, 제1타겟 행렬에 대한 연산을 수행한다. 예컨대, 연산부(530)는 데이터 로딩부(520)에 의해 로딩된 또다른 제2타겟 행렬의 엘리먼트값과, 제1타겟 행렬의 논제로 엘리먼트값에 대한 연산을 수행할 수 있다. 제2타겟 행렬은 제2메모리(550)에 저장될 수 있으며, 실시예에 따라서, 제2타겟 행렬의 모든 엘리먼트값이 제2메모리(550)에 저장되거나 제1타겟 행렬과 같이 행렬 인덱스 정보 형태로 제2메모리(550)에 저장될 수 있다.

[0053] 또한 일예로서, 제1타겟 행렬은 인공 신경망의 가중치값을 포함하는 가중치 행렬일 수 있으며, 제2타겟 행렬은 가중치값의 활성화 여부를 결정하는 엘리먼트를 포함하는 행렬일 수 있다. 즉, 제2타겟 행렬은 활성화 함수의 역할을 수행하는 행렬일 수 있다. 또는 실시예에 따라서, 제1타겟 행렬은 제1레이어에 대한 가중치 행렬일 수 있으며, 제2타겟 행렬은 제2레이어에 대한 가중치 행렬일 수 있다.

[0054] 연산부(530)는 병렬 연산을 위한 복수의 연산기(Processing element)를 포함할 수 있으며, 연산기 각각에 제1타겟 행렬의 논제로 엘리먼트값이 할당될 수 있다. 연산기 각각은 할당된 제1타겟 행렬의 논제로 엘리먼트값과 제2타겟 행렬의 엘리먼트에 대한 연산을 수행할 수 있다.

[0056] 도 6은 본 발명의 일실시예에 따른 행렬 인덱스 정보를 이용하는 행렬 처리 방법을 설명하기 위한 도면이다.

[0057] 도 6을 참조하면, 본 발명의 일실시예에 따른 행렬 처리 장치는 제1타겟 행렬에 대한 행렬 인덱스 정보를 이용하여, 제1타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩(S610)하고, 로딩된 데이터를 연산기로 전달(S620)한다. 여기서, 행렬 인덱스 정보는 전술된 실시예에서 생성된 행렬 인덱스 정보와 같이, 제1타겟 행렬의 논제로 엘리먼트에 대한 개수 정보와, 제1타겟 행렬내에서 논제로 엘리먼트의 위치 정보를 포함한다.

[0058] 메모리에는 행렬 인덱스 정보 및 타겟 행렬의 논제로 엘리먼트값이 저장되며, 서로 다른 크기의 타겟 행렬의 행렬 인덱스 정보와 논제로 엘리먼트값이 저장될 수도 있다. 이 경우, 서로 다른 행렬 인덱스 정보에는 대응되는 타겟 행렬의 크기 정보가 더 포함될 수 있다. 타겟 행렬의 크기 정보는, 타겟 행렬의 행과 열의 크기를 나타내는 인덱스로 표현될 수 있다.

[0059] 단계 S610에서 행렬 처리 장치는 제1타겟 행렬의 행렬 인덱스 정보를 이용하여, 제2타겟 행렬의 엘리먼트 중에서, 제1타겟 행렬의 논제로 엘리먼트와의 곱셈 대상인 엘리먼트를 메모리에서 로딩할 수 있다. 그리고 로딩된 제2타겟 행렬의 엘리먼트는 단계 S620에서 연산기로 전달되어, 제1타겟 행렬과의 곱셈 연산에 이용될 수 있다.

[0060] 제2타겟 행렬의 모든 엘리먼트는 메모리에 저장될 수 있는데, 제1타겟 행렬의 제로 엘리먼트와 곱해지는 제2타겟 행렬의 엘리먼트를 로딩하는 것은 불필요하므로, 행렬 처리 장치는 제1타겟 행렬의 논제로 엘리먼트와 곱셈이 이루어지는 제2타겟 행렬의 엘리먼트를 선택적으로 메모리에서 로딩할 수 있다.

[0061] 예컨대, 제1타겟 행렬의 논제로 엘리먼트의 개수가 1개이며, 그 위치가 제1행, 제1열에 대응된다면, 행렬 처리 장치는 제2타겟 행렬의 엘리먼트 중에서, 제1행, 제1열에 위치하는 엘리먼트를 로딩할 수 있다.

[0062] 한편, 실시예에 따라서 행렬 처리 장치는 단계 S610에서, 제3타겟 행렬에 대한 행렬 인덱스 정보를 이용하여, 제3타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩할 수 있다. 이 경우, 행렬 처리 장치는 단계 S620에서 로딩된 논제로 엘리먼트값뿐만 아니라, 제1 및 제3타겟 행렬에 대한 행렬 인덱스 정보를 함께 연산기로 전달할 수 있다.

[0063] 또는 실시예에 따라서 행렬 처리 장치는 단계 S620에서 행렬 인덱스 정보 및 제1타겟 행렬의 논제로 엘리먼트값을 이용하여, 제1타겟 행렬을 복원하고, 복원된 제1타겟 행렬을 연산기로 전달할 수 있다. 행렬 처리 장치는 행렬 인덱스 정보를 통해 제1타겟 행렬의 제로 엘리먼트의 위치를 확인할 수 있으며, 제로 엘리먼트의 위치에 제로를 패딩(padding)함으로서, 제1타겟 행렬을 복원할 수 있다.

[0065] 도 7은 메모리에 저장된 행렬 인덱스 정보의 일예를 나타내는 도면이다.

[0066] 본 발명의 일실시예에 따른 행렬 처리 장치는 단계 S610에서 제1타겟 행렬의 논제로 엘리먼트값에 할당된 메모

리 주소값을 이용하여, 제1타겟 행렬의 논제로 엘리먼트값을 로딩할 수 있다. 행렬 처리 장치는 행렬 인덱스 정보를 이용하여, 제1타겟 행렬의 논제로 엘리먼트값에 대한 주소값을 결정하고, 결정된 주소값을 이용하여, 제1타겟 행렬의 논제로 엘리먼트값을 로딩할 수 있다.

[0067] 전술된 바와 같이, 논제로 엘리먼트값에 할당된 메모리 주소값은, 미리 설정된 규칙에 따라 연속된 형태일 수 있으며, 이 경우 행렬 처리 장치는 제1타겟 행렬의 논제로 엘리먼트값보다 이전에 메모리에서 로딩된 논제로 엘리먼트값의 개수를 이용하여, 제1타겟 행렬의 논제로 엘리먼트값에 대한 주소값을 결정할 수 있다.

[0068] 예컨대, 도 7과 같이 제1 및 제2행렬 인덱스 정보(710, 720)와 논제로 엘리먼트값(730)이 메모리에 저장된 상태에서, 제1행렬 인덱스 정보(710)을 통해, 제1타겟 행렬보다 이전에 로딩된 논제로 엘리먼트값(0.1, 0.25) 2개에 대한 메모리 주소값이 N, N+1이라면, 행렬 처리 장치는 제2행렬 인덱스 정보(720)를 이용하여, 제1타겟 행렬의 논제로 엘리먼트값 3개에 대한 메모리 주소값을 각각, N+2, N+3, N+4로 결정할 수 있다. 따라서, 행렬 처리 장치는 메모리 주소값 N+2, N+3, N+4에 대응되는 제1타겟 행렬의 논제로 엘리먼트 -0.5, -0.25, 0.5를 메모리로부터 로딩할 수 있다.

[0069] 본 발명의 일실시예에 따른 행렬 처리 장치는 버스트 모드(burst mode)를 이용하여, 논제로 엘리먼트값을 효율적으로 메모리로부터 로딩할 수 있다.

[0071] 도 8은 본 발명의 다른 실시예에 따른 행렬 인덱스 정보를 이용하는 행렬 처리 방법을 설명하기 위한 도면이다.

[0072] 도 8을 참조하면, 본 발명의 일실시예에 따른 행렬 처리 장치는, 단계 S610에서 로딩된 논제로 엘리먼트의 개수와 연산기의 개수를 비교(S810)한다. 그리고 비교 결과에 따라서, 로딩된 논제로 엘리먼트값을 연산기로 전달(S820)한다.

[0073] 행렬 처리 장치는, 단계 S610에서 로딩된 논제로 엘리먼트값의 개수가 연산기의 개수보다 적은 경우, 로딩된 논제로 엘리먼트값을 바로 연산기로 전달하지 않고, 단계 S820에서, 제1타겟 행렬의 논제로 엘리먼트값 이후에 메모리에서 로딩되는 논제로 엘리먼트값을, 제1타겟 행렬의 논제로 엘리먼트값과 함께 연산기로 전달한다.

[0074] 예컨대, 연산기의 개수가 6개이며, 제1시점에서 로딩된 제1타겟 행렬의 논제로 엘리먼트값이 3개라면, 행렬 처리 장치는, 제1타겟 행렬의 논제로 엘리먼트값을 바로 연산기로 전달하는 것이 아니라, 제1시점이 이후인 제2시점에서 새로운 논제로 엘리먼트값이 로딩되면, 새로운 논제로 엘리먼트값과 함께 제1타겟 행렬의 논제로 엘리먼트값을 연산기로 전달한다.

[0075] 행렬 연산은 여러 연산기에서 병렬 처리되므로, 연산기의 개수에 가까운 개수만큼의 논제로 엘리먼트에 대한 값이 한번에 연산기로 전달될 경우 연산기의 사용 효율이 높아질 수 있다. 따라서, 본 발명의 일실시예에 따른 행렬 처리 장치는 로딩된 논제로 엘리먼트의 개수와 연산기의 개수를 비교하고, 로딩된 논제로 엘리먼트의 개수가 연산기의 개수보다 적은 경우, 로딩된 논제로 엘리먼트를 누적시켜 한번에 연산기로 전달하여 연산기의 사용 효율을 높인다.

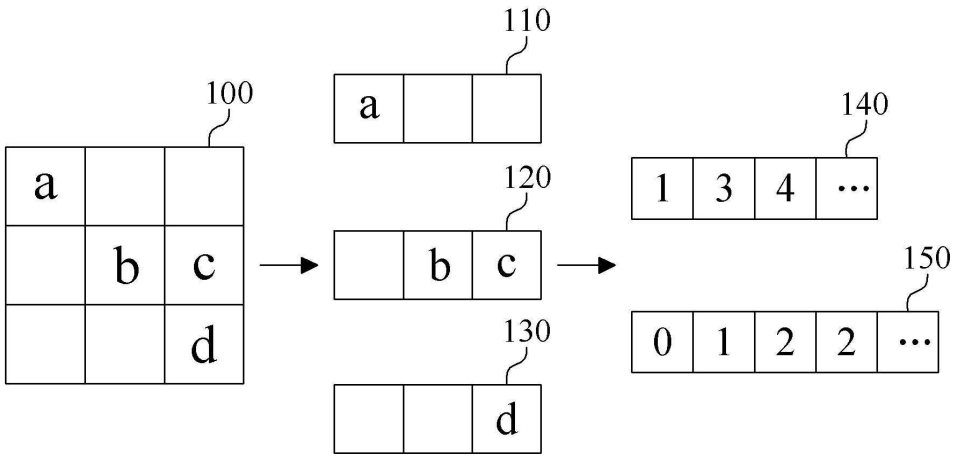
[0077] 앞서 설명한 기술적 내용들은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예들을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 하드웨어 장치는 실시예들의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

[0079] 이상과 같이 본 발명에서는 구체적인 구성 요소 등과 같은 특정 사항들과 한정된 실시예 및 도면에 의해 설명되었으나 이는 본 발명의 보다 전반적인 이해를 돕기 위해서 제공된 것일 뿐, 본 발명은 상기의 실시예에 한정되는 것은 아니며, 본 발명이 속하는 분야에서 통상적인 지식을 가진 자라면 이러한 기재로부터 다양한 수정 및 변형이 가능하다. 따라서, 본 발명의 사상은 설명된 실시예에 국한되어 정해져서는 아니되며, 후술하는 특허청구범위뿐 아니라 이 특허청구범위와 균등하거나 등가적 변형이 있는 모든 것들은 본 발명 사상의 범주에 속한다

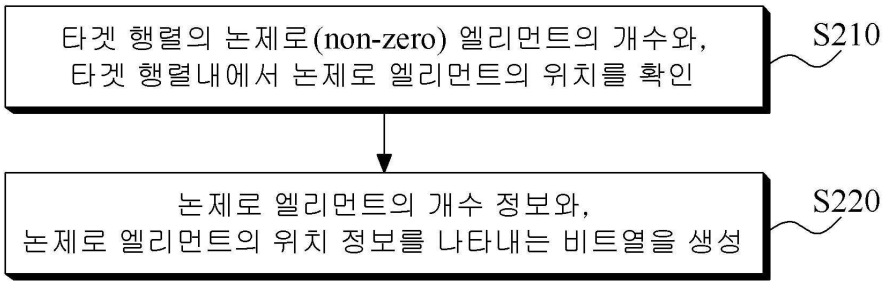
고 할 것이다.

도면

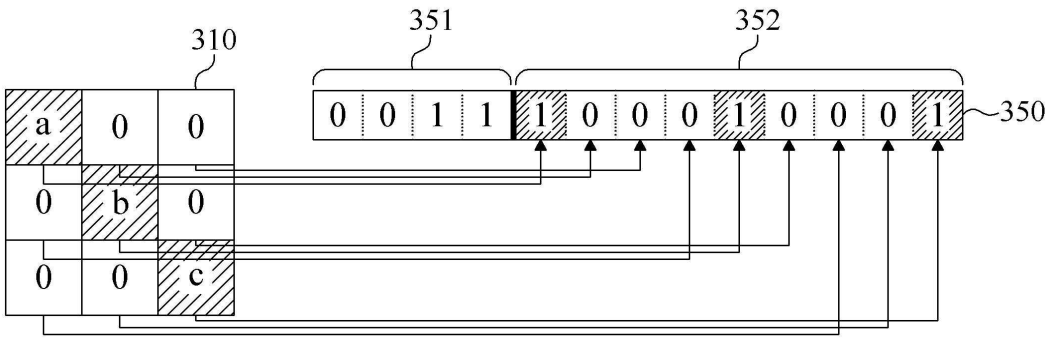
도면1



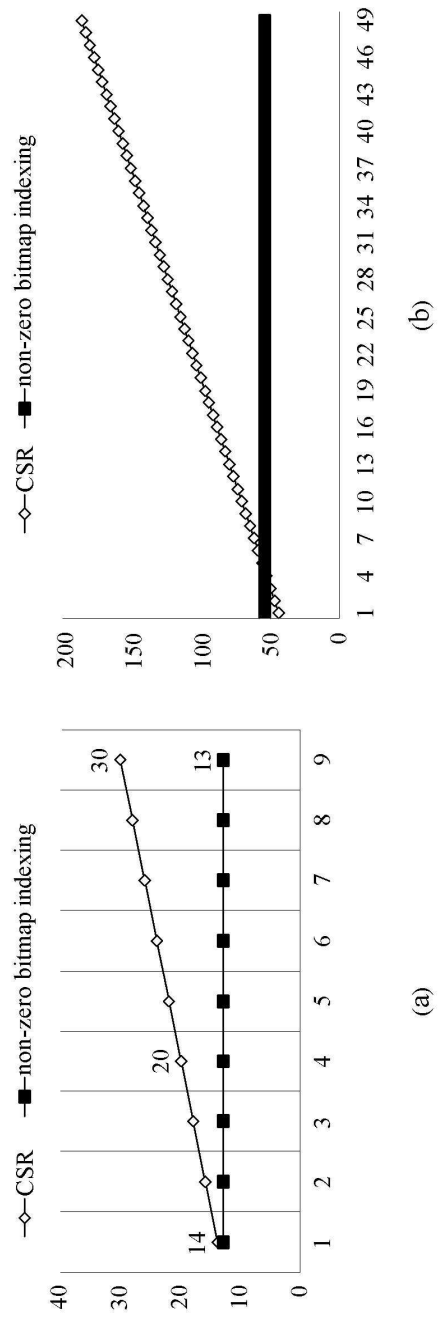
도면2



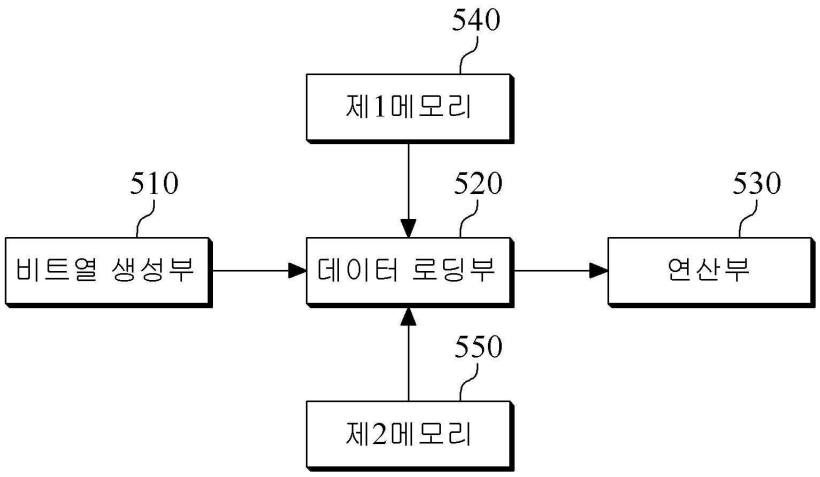
도면3



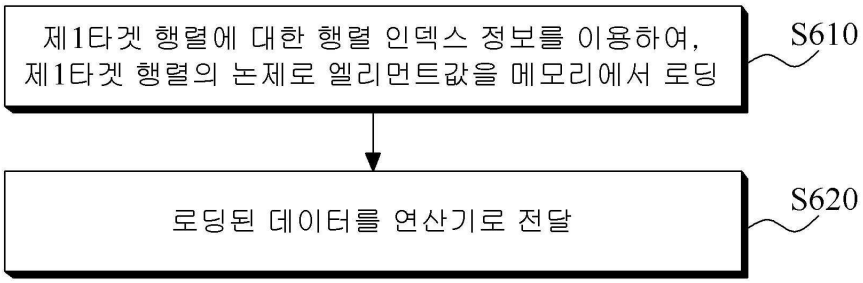
도면4


CSR
non-zero bitmap indexing
40
30
20
10
0
1
2
3
4
5
6
7
8
9
30
20
14
13
(a)
CSR
non-zero bitmap indexing
200
150
100
50
0
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49
(b)

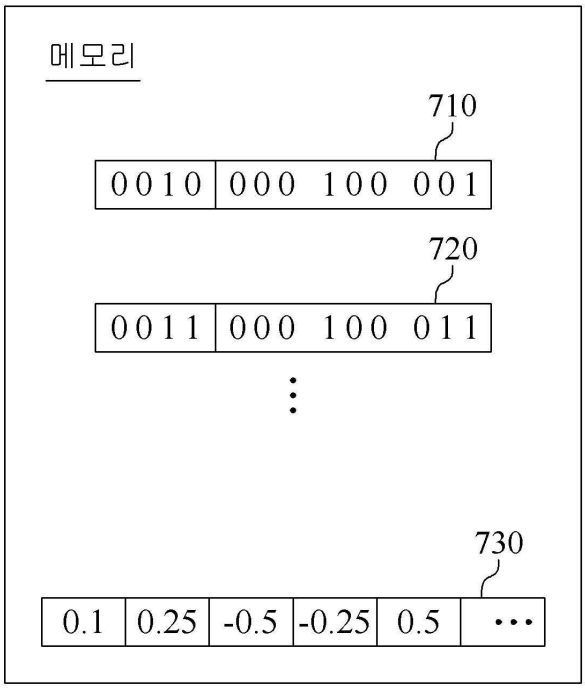
도면5


540
제1메모리
510
520
530
비트열 생성부
데이터 로딩부
연산부
550
제2메모리

도면6


제1타겟 행렬에 대한 행렬 인덱스 정보를 이용하여,
제1타겟 행렬의 논제로 엘리먼트값을 메모리에서 로딩
S610
로딩된 데이터를 연산기로 전달
S620

도면7


메모리
710
0010
000 100 001
720
0011
000 100 011
730
0.1
0.25
-0.5
-0.25
0.5
•••

도면8