(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0091660
(43) 공개일자 2025년06월23일

(51) 국제특허분류(Int. Cl.)
G06N 3/065 (2023.01) *G06N 3/042* (2023.01)
G06N 3/0499 (2023.01) *G06N 3/08* (2023.01)
(52) CPC특허분류
G06N 3/065 (2023.01)
G06N 3/042 (2023.01)
(21) 출원번호 10-2023-0181829
(22) 출원일자 2023년12월14일
심사청구일자 2023년12월14일

(71) 출원인
울산과학기술원
울산광역시 울주군 언양읍 유니스트길 50
(72) 발명자
최영리
울산광역시 울주군 언양읍 유니스트길 50
강준수
울산광역시 울주군 언양읍 유니스트길 50
(74) 대리인
특허법인비엘티

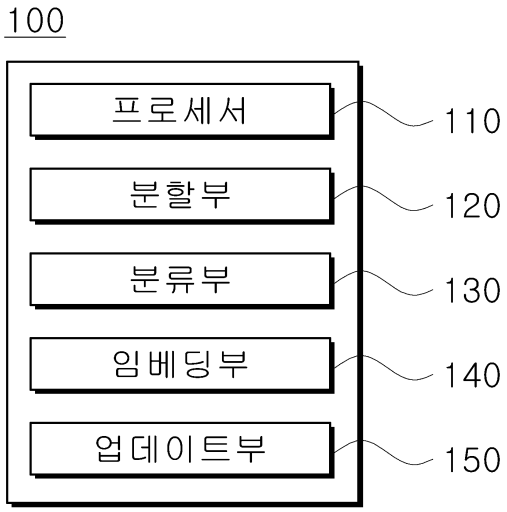
전체 청구항 수 : 총 18 항

(54) 발명의 명칭 **서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치 및 방법**

(57) 요 약

본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치는 학습 데이터셋을 배치 사이즈 단위의 서브그래프로 분할하는 분할부; 각 서브그래프에 포함된 노드를, 외부 서브그래프의 노드와 인접하게 배치된 경계 노드와 상기 서브그래프 내부에 위치한 노드만 인접하게 배치된 내부 노드로 분류하는 분류부; 상기 각 서브그래프의 중간 임베딩(intermediate embedding)으로서 상기 경계 노드에 대한 임베딩을 저장 및 관리하는 임베딩부; 및 상기 분할부, 상기 분류부 및 상기 임베딩부를 제어하는 적어도 하나의 프로세서를 포함하며, 상기 적어도 하나의 프로세서는, 그래프 데이터셋을 각 서브그래프로 분할하는 도중, 상기 각 서브그래프에 포함된 노드를 경계 노드와 내부 노드로 분류하고, 상기 경계 노드에 대한 중간 임베딩만을 저장하여 중간 임베딩 테이블로 관리하도록 구성될 수 있다.

대 표 도 - 도1



(52) CPC특허분류
G06N 3/0499 (2023.01)
G06N 3/084 (2023.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 1711193910
과제번호 2021-0-01817-003
부처명 과학기술정보통신부
과제관리(전문)기관명 정보통신기획평가원
연구사업명 정보통신방송혁신인재양성
연구과제명 하이퍼-컴포저블 데이터센터를 위한 차세대 컴퓨팅 기술 개발
기 여 율 90/100
과제수행기관명 울산과학기술원
연구기간 2023.01.01 ~ 2023.12.31
이 발명을 지원한 국가연구개발사업
과제고유번호 1711181473
과제번호 2023R1A2C2006432
부처명 과학기술정보통신부
과제관리(전문)기관명 한국연구재단
연구사업명 개인기초연구(과기정통부)
연구과제명 확장 가능한 적응형 고성능 분리형 스토리지 시스템 기술 연구
기 여 율 10/100
과제수행기관명 울산과학기술원
연구기간 2023.03.01 ~ 2024.02.29

명 세 서

청구범위

청구항 1

학습 데이터셋을 배치 사이즈 단위의 서브그래프로 분할하는 분할부;

각 서브그래프에 포함된 노드를, 외부 서브그래프의 노드와 인접하게 배치된 경계 노드와 상기 서브그래프 내부에 위치한 노드만 인접하게 배치된 내부 노드로 분류하는 분류부;

상기 각 서브그래프의 중간 임베딩(intermediate embedding)으로서 상기 경계 노드에 대한 임베딩을 저장 및 관리하는 임베딩부; 및

상기 분할부, 상기 분류부 및 상기 임베딩부를 제어하는 적어도 하나의 프로세서를 포함하며,

상기 적어도 하나의 프로세서는,

그래프 데이터셋을 각 서브그래프로 분할하는 도중, 상기 각 서브그래프에 포함된 노드를 경계 노드와 내부 노드로 분류하고,

상기 경계 노드에 대한 중간 임베딩만을 저장하여 중간 임베딩 테이블로 관리하도록 구성된

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치.

청구항 2

제1 항에 있어서,

기설정된 간격(staleness) 별로 상기 중간 임베딩을 업데이트하는 업데이트부를 더 포함하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치.

청구항 3

제2 항에 있어서,

상기 적어도 하나의 프로세서는,

상기 업데이트부가 각 학습 단계에 참여한 서브그래프와 인접한 외부 서브그래프의 중간 임베딩만을 업데이트하도록 구성된

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치.

청구항 4

제3 항에 있어서,

상기 적어도 하나의 프로세서는,

상기 업데이트부가 상기 인접한 외부 서브그래프의 중간 임베딩이 업데이트된 이전 학습 단계를 확인하고,

상기 인접한 외부 서브그래프의 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초과한 서브그래프만 상기 중간 임베딩을 업데이트하도록 구성된

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치.

청구항 5

제4 항에 있어서,

상기 적어도 하나의 프로세서는,

각 학습 단계마다 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초과한 상기 인접한 외부 서

브그래프의 개수가 최소인 서브그래프를 배치로 선택하도록 구성된

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치.

청구항 6

제5 항에 있어서,

상기 적어도 하나의 프로세서는,

상기 배치로 선택된 서브그래프의 인덱스를 메인 학습 모듈에 전달하고,

상기 메인 학습 모델이 상기 서브그래프의 인덱스와 상기 서브그래프에 인접한 외부 서브그래프의 중간 임베딩을 로드하여 GPU에서 순전파 및 역전파 학습을 진행하기 위한 상기 배치를 생성하도록 구성된

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치.

청구항 7

제1 항에 있어서

상기 중간 임베딩 테이블은 이전 임베딩 테이블보다 메모리 크기가 작은 것을 특징으로 하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치.

청구항 8

제7 항에 있어서,

상기 적어도 하나의 프로세서는,

상기 이전 임베딩 테이블을 상기 중간 임베딩 테이블로 업데이트하는 경우, 상기 이전 임베딩 테이블의 상기 서브그래프 주소를 상기 중간 임베딩 테이블의 상기 서브그래프 주소로 리맵핑하도록 구성된

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치.

청구항 9

학습을 위한 그래프 데이터셋을 배치 사이즈 단위의 서브그래프로 분할하는 단계;

각 서브그래프에 포함된 노드를, 외부 서브그래프의 노드와 인접하게 배치된 경계 노드와 상기 서브그래프 내부에 위치한 노드만 인접하게 배치된 내부 노드로 분류하는 단계; 및

상기 경계 노드에 대한 중간 임베딩만을 상기 각 서브그래프의 중간 임베딩 테이블로 저장 및 관리하는 단계를 포함하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법.

청구항 10

제9 항에 있어서,

기설정된 간격(staleness) 별로 상기 중간 임베딩을 업데이트하는 단계를 더 포함하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법.

청구항 11

제10 항에 있어서,

상기 업데이트하는 단계는,

각 학습 단계에 참여한 서브그래프와 인접한 외부 서브그래프의 중간 임베딩만을 업데이트하는 단계를 포함하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법.

청구항 12

제11 항에 있어서,

상기 업데이트하는 단계는,

상기 인접한 외부 서브그래프의 중간 임베딩이 업데이트된 이전 학습 단계를 확인하는 단계; 및

상기 인접한 외부 서브그래프의 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초과한 서브그래프만 상기 중간 임베딩을 업데이트하는 단계를 더 포함하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법.

청구항 13

제12 항에 있어서,

상기 업데이트하는 단계는,

각 학습 단계마다 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초과한 상기 인접한 외부 서브그래프의 개수가 최소인 서브그래프를 배치로 선택하는 단계를 더 포함하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법.

청구항 14

제13 항에 있어서,

상기 배치로 선택되는 서브그래프는 하나의 에포크(epoch)에서 배치로 선택되지 않은 서브그래프인 것을 특징으로 하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법.

청구항 15

제14 항에 있어서,

상기 배치로 선택된 서브그래프의 인덱스를 메인 학습 모듈에 전달하는 단계;

상기 메인 학습 모델이 상기 서브그래프의 인덱스와 상기 서브그래프에 인접한 외부 서브그래프의 중간 임베딩을 로드하여 상기 배치를 생성하는 단계; 및

상기 배치를 이용하여 GPU에서 순전파 및 역전파 학습을 진행하는 단계를 더 포함하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법.

청구항 16

제9 항에 있어서,

상기 중간 임베딩 테이블은 이전 임베딩 테이블보다 메모리 크기가 작은 것을 특징으로 하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법.

청구항 17

제16 항에 있어서,

상기 이전 임베딩 테이블을 상기 중간 임베딩 테이블로 업데이트하는 경우, 상기 이전 임베딩 테이블의 상기 서브그래프 주소를 상기 중간 임베딩 테이블의 상기 서브그래프 주소로 리맵핑하는 단계를 더 포함하는

서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법.

청구항 18

컴퓨터와 결합되어 제9 항 내지 제17 항 중 어느 한 항의 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모

리 관리 방법을 실행시키기 위하여 컴퓨터 판독 가능한 기록매체에 저장된 프로그램.

발명의 설명

기 술 분 야

[0001] 본 발명은 그래프 뉴럴 네트워크의 딥러닝 메모리 최적화 장치 및 방법에 관한 것으로, 보다 구체적으로 그래프 뉴럴 네트워크의 학습 과정에서 불필요한 메모리를 제거하고 중간 임베딩의 업데이트 비용을 최소화하는 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치 및 방법에 관한 것이다.

배 경 기 술

[0002] 최근 연구에서 그래프 뉴럴 네트워크 학습 과정 중 학습 장치인 GPU와 상기 GPU의 호스트 메모리 사이의 커뮤니케이션 비용을 줄이기 위해 학습 데이터셋을 배치 사이즈 단위의 서브그래프로 분할하는 기술이 제시되었다.

[0003] 그러나, 분할 도중 각 서브그래프와 인접해있던 다른 서브그래프와의 관계를 나타내는 노드 또는 에지들을 잃어 학습 과정 중 정보 손실이 발생하게 되고, 학습 정확도가 떨어지는 문제점이 발생하였다.

[0004] 종래에는 이를 해결하기 위해 학습 단계마다 각 서브그래프의 인접한 다른 서브그래프의 에지 또는 노드들을 포함하여 배칭하였으나, 인접한 다른 서브그래프의 에지 또는 노드를 같이 배칭하는 것은 학습 장치인 GPU의 메모리 제한으로 물리적으로 구현하기 어려웠다.

[0005] 특히, 그래프 뉴럴 네트워크 구조 특성상 모델의 레이어 개수마다 재귀적으로 인접해 있는 에지 또는 노드들을 전부 포함시켜야 하여 연산 및 메모리가 모델의 레이어 개수에 따라 지수적으로 증가하게되어 한정된 메모리로는 이를 모두 저장할 수 없는 문제점이 존재하였다.

선행기술문헌

비특허문헌

[0006] (비특허문헌 0001) Xin Liu 외 5인, “Sampling methods for efficient training graph convolutional networks: A survey”,arXiv:2103.05872v3, 2021년 9월 3일

발명의 내용

해결하려는 과제

[0007] 본 발명의 실시예는 상술한 문제점을 해결하기 위하여 제안된 것으로, 그래프 뉴럴 네트워크의 학습 과정에서 불필요한 메모리를 제거하고 중간 임베딩의 업데이트 비용을 최소화하는 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치 및 방법을 제공할 수 있다.

[0008] 본 발명이 해결하고자 하는 과제들은 이상에서 언급된 과제로 제한되지 않으며, 언급되지 않은 또 다른 과제들은 아래의 기재로부터 통상의 기술자에게 명확하게 이해될 수 있을 것이다.

과제의 해결 수단

[0009] 상술한 과제를 해결하기 위하여, 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치는 학습 데이터셋을 배치 사이즈 단위의 서브그래프로 분할하는 분할부; 각 서브그래프에 포함된 노드를, 외부 서브그래프의 노드와 인접하게 배치된 경계 노드와 상기 서브그래프 내부에 위치한 노드만 인접하게 배치된 내부 노드로 분류하는 분류부; 상기 각 서브그래프의 중간 임베딩(intermediate embedding)으로서 상기 경계 노드에 대한 임베딩을 저장 및 관리하는 임베딩부; 및 상기 분할부, 상기 분류부 및 상기 임베딩부를 제어하는 적어도 하나의 프로세서를 포함하며, 상기 적어도 하나의 프로세서는, 그래프 데이터셋을 각 서브그래프로 분할하는 도중, 상기 각 서브그래프에 포함된 노드를 경계 노드와 내부 노드로 분류하고, 상기 경계 노드에 대한 중간 임베딩만을 저장하여 중간 임베딩 테이블로 관리하도록 구성될 수 있다.

[0010] 상술한 과제를 해결하기 위하여, 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법은 학습을 위한 그래프 데이터셋을 배치 사이즈 단위의 서브그래프로 분할하는 단계; 각 서브

그래프에 포함된 노드를, 외부 서브그래프의 노드와 인접하게 배치된 경계 노드와 상기 서브그래프 내부에 위치한 노드만 인접하게 배치된 내부 노드로 분류하는 단계; 및 상기 경계 노드에 대한 중간 임베딩만을 상기 각 서브그래프의 중간 임베딩 테이블로 저장 및 관리하는 단계를 포함할 수 있다.

[0011] 이 외에도, 본 발명을 구현하기 위한 다른 방법, 다른 시스템 및 상기 방법을 실행하기 위한 컴퓨터 프로그램을 기록하는 컴퓨터로 판독 가능한 기록 매체가 더 제공될 수 있다.

[0012] 이 외에도, 본 발명을 구현하기 위한 방법이 컴퓨터 상에서 수행되도록 매체에 저장되는 컴퓨터 프로그램이 더 제공될 수 있다.

발명의 효과

[0013] 본 발명의 전술한 과제 해결 수단에 의하면, 호스트 메모리 크기를 줄이고, 불필요한 중간 임베딩 업데이트 없이 최적의 중간 임베딩 업데이트를 수행하며, 업데이트 과정에서 요구되는 임베딩의 개수를 최소화하여 학습 시간을 최적화하는 효과를 제공할 수 있다.

[0014] 본 발명의 효과들은 이상에서 언급된 효과로 제한되지 않으며, 언급되지 않은 또 다른 효과들은 아래의 기재로부터 통상의 기술자에게 명확하게 이해될 수 있다.

도면의 간단한 설명

[0015] 도 1은 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치의 구성을 간략하게 도시한 블록도이다.

도 2는 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치의 서브그래프를 도시한 개념도이다.

도 3은 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치의 학습 과정에서 학습 단계 별로 선택되는 배치를 도시한 개념도이다.

도 4는 종래 그래프 뉴럴 네트워크의 메모리 관리 전자 장치의 학습 과정에서 중간 임베딩 업데이트를 수행하는 배치와 본 발명의 일 실시예에 따른 전자 장치의 학습 과정에서 중간 임베딩 업데이트를 수행하는 배치를 도시한 개념도이다.

도 5는 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치의 메인 학습 모듈의 학습 과정을 도시한 프로세스도이다.

도 6은 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법의 플로우 차트를 도시한 것이다.

발명을 실시하기 위한 구체적인 내용

[0016] 본 발명 전체에 걸쳐 동일 참조 부호는 동일 구성요소를 지칭한다. 본 발명이 실시예들의 모든 요소들을 설명하는 것은 아니며, 본 발명이 속하는 기술분야에서 일반적인 내용 또는 실시예들 간에 중복되는 내용은 생략한다. 명세서에서 사용되는 '부, 모듈, 부재, 블록'이라는 용어는 소프트웨어 또는 하드웨어로 구현될 수 있으며, 실시예들에 따라 복수의 '부, 모듈, 부재, 블록'이 하나의 구성요소로 구현되거나, 하나의 '부, 모듈, 부재, 블록'이 복수의 구성요소들을 포함하는 것도 가능하다.

[0017] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 직접적으로 연결되어 있는 경우뿐 아니라, 간접적으로 연결되어 있는 경우를 포함하고, 간접적인 연결은 무선 통신망을 통해 연결되는 것을 포함한다.

[0018] 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.

[0019] 명세서 전체에서, 어떤 부재가 다른 부재 "상에" 위치하고 있다고 할 때, 이는 어떤 부재가 다른 부재에 접해 있는 경우뿐 아니라 두 부재 사이에 또 다른 부재가 존재하는 경우도 포함한다.

[0020] 제 1, 제 2 등의 용어는 하나의 구성요소를 다른 구성요소로부터 구별하기 위해 사용되는 것으로, 구성요소가 전술된 용어들에 의해 제한되는 것은 아니다.

[0021] 단수의 표현은 문맥상 명백하게 예외가 있지 않는 한, 복수의 표현을 포함한다.

[0022] 각 단계들에 있어 식별부호는 설명의 편의를 위하여 사용되는 것으로 식별부호는 각 단계들의 순서를 설명하는 것이 아니며, 각 단계들은 문맥상 명백하게 특정 순서를 기재하지 않는 이상 명기된 순서와 다르게 실시될 수 있다.

[0023] 이하 첨부된 도면들을 참고하여 본 발명의 작용 원리 및 실시예들에 대해 설명한다.

[0024] 본 명세서에서 '본 발명에 따른 장치'는 연산처리를 수행하여 사용자에게 결과를 제공할 수 있는 다양한 장치들이 모두 포함된다. 예를 들어, 본 발명에 따른 장치는, 컴퓨터, 서버 장치 및 휴대용 단말기를 모두 포함하거나, 또는 어느 하나의 형태가 될 수 있다.

[0025] 여기에서, 상기 컴퓨터는 예를 들어, 웹 브라우저(WEB Browser)가 탑재된 노트북, 데스크톱(desktop), 랩톱(laptop), 태블릿 PC, 슬레이트 PC 등을 포함할 수 있다.

[0026] 상기 서버 장치는 외부 장치와 통신을 수행하여 정보를 처리하는 서버로써, 애플리케이션 서버, 컴퓨팅 서버, 데이터베이스 서버, 파일 서버, 게임 서버, 메일 서버, 프록시 서버 및 웹 서버 등을 포함할 수 있다.

[0027] 상기 휴대용 단말기는 예를 들어, 휴대성과 이동성이 보장되는 무선 통신 장치로서, PCS(Personal Communication System), GSM(Global System for Mobile communications), PDC(Personal Digital Cellular), PHS(Personal Handyphone System), PDA(Personal Digital Assistant), IMT(International Mobile Telecommunication)-2000, CDMA(Code Division Multiple Access)-2000, W-CDMA(W-Code Division Multiple Access), WiBro(Wireless Broadband Internet) 단말, 스마트 폰(Smart Phone) 등과 같은 모든 종류의 핸드헬드(Handheld) 기반의 무선 통신 장치와 시계, 반지, 팔찌, 발찌, 목걸이, 안경, 콘택트 렌즈, 또는 머리 착용형 장치(head-mounted-device(HMD) 등과 같은 웨어러블 장치를 포함할 수 있다.

[0028] 본 발명에 따른 인공지능과 관련된 기능은 프로세서와 메모리를 통해 동작된다. 프로세서는 하나 또는 복수의 프로세서로 구성될 수 있다. 이때, 하나 또는 복수의 프로세서는 CPU, AP, DSP(Digital Signal Processor) 등과 같은 범용 프로세서, GPU, VPU(Vision Processing Unit)와 같은 그래픽 전용 프로세서 또는 NPU와 같은 인공지능 전용 프로세서일 수 있다. 하나 또는 복수의 프로세서는, 메모리에 저장된 기 정의된 동작 규칙 또는 인공 지능 모델에 따라, 입력 데이터를 처리하도록 제어한다. 또는, 하나 또는 복수의 프로세서가 인공지능 전용 프로세서인 경우, 인공지능 전용 프로세서는, 특정 인공 지능 모델의 처리에 특화된 하드웨어 구조로 설계될 수 있다.

[0029] 기 정의된 동작 규칙 또는 인공 지능 모델은 학습을 통해 만들어진 것을 특징으로 한다. 여기서, 학습을 통해 만들어진다는 것은, 기본 인공 지능 모델이 학습 알고리즘에 의하여 다수의 학습 데이터들을 이용하여 학습됨으로써, 원하는 특성(또는, 목적)을 수행하도록 설정된 기 정의된 동작 규칙 또는 인공 지능 모델이 만들어짐을 의미한다. 이러한 학습은 본 발명에 따른 인공지능이 수행되는 기기 자체에서 이루어질 수도 있고, 별도의 서버 및/또는 시스템을 통해 이루어 질 수도 있다. 학습 알고리즘의 예로는, 지도형 학습(supervised learning), 비지도 형 학습(unsupervised learning), 준지도형 학습(semi-supervised learning) 또는 강화 학습(reinforcement learning)이 있으나, 전술한 예에 한정되지 않는다.

[0030] 인공 지능 모델은, 복수의 신경망 레이어들로 구성될 수 있다. 복수의 신경망 레이어들 각각은 복수의 가중치들(weight values)을 갖고 있으며, 이전(previous) 레이어의 연산 결과와 복수의 가중치들 간의 연산을 통해 신경망 연산을 수행한다. 복수의 신경망 레이어들이 갖고 있는 복수의 가중치들은 인공 지능 모델의 학습 결과에 의해 최적화될 수 있다. 예를 들어, 학습 과정 동안 인공 지능 모델에서 획득한 로스(loss) 값 또는 코스트(cost) 값이 감소 또는 최소화되도록 복수의 가중치들이 갱신될 수 있다. 인공 신경망은 심층 신경망(DNN:Deep Neural Network)를 포함할 수 있으며, 예를 들어, CNN (Convolutional Neural Network), DNN (Deep Neural Network), RNN (Recurrent Neural Network), RBM (Restricted Boltzmann Machine), DBN (Deep Belief Network), BRDNN(Bidirectional Recurrent Deep Neural Network) 또는 심층 Q-네트워크 (Deep Q-Networks) 등이 있으나, 전술한 예에 한정되지 않는다.

[0031] 본 발명의 예시적인 실시예에 따르면, 프로세서는 인공지능을 구현할 수 있다. 인공지능이란 사람의 신경세포(biological neuron)를 모사하여 기계가 학습하도록 하는 인공신경망(Artificial Neural Network) 기반의 기계학습법을 의미한다. 인공지능의 방법론에는 학습 방식에 따라 훈련데이터로서 입력데이터와 출력데이터가 같이 제공됨으로써 문제(입력데이터)의 해답(출력데이터)이 정해져 있는 지도학습(supervised learning), 및 출력데

이터 없이 입력데이터만 제공되어 문제(입력데이터)의 해답(출력데이터)이 정해지지 않는 비지도학습(unsupervised learning), 및 현재의 상태(State)에서 어떤 행동(Action)을 취할 때마다 외부 환경에서 보상(Reward)이 주어지는데, 이러한 보상을 최대화하는 방향으로 학습을 진행하는 강화학습(reinforcement learning)으로 구분될 수 있다. 또한, 인공지능의 방법론은 학습 모델의 구조인 아키텍처에 따라 구분될 수도 있는데, 널리 이용되는 딥러닝 기술의 아키텍처는, 합성곱신경망(CNN; Convolutional Neural Network), 순환신경망(RNN; Recurrent Neural Network), 트랜스포머(Transformer), 생성적 대립 신경망(GAN; generative adversarial networks) 등으로 구분될 수 있다.

[0032] 본 장치와 시스템은 인공 지능 모델을 포함할 수 있다. 인공 지능 모델은 하나의 인공 지능 모델일 수 있고, 복수의 인공 지능 모델로 구현될 수도 있다. 인공 지능 모델은 뉴럴 네트워크(또는 인공 신경망)로 구성될 수 있으며, 기계학습과 인지과학에서 생물학의 신경을 모방한 통계학적 학습 알고리즘을 포함할 수 있다. 뉴럴 네트워크는 시냅스의 결합으로 네트워크를 형성한 인공 뉴런(노드)이 학습을 통해 시냅스의 결합 세기를 변화시켜, 문제 해결 능력을 가지는 모델 전반을 의미할 수 있다. 뉴럴 네트워크의 뉴런은 가중치 또는 바이어스의 조합을 포함할 수 있다. 뉴럴 네트워크는 하나 이상의 뉴런 또는 노드로 구성된 하나 이상의 레이어(layer)를 포함할 수 있다. 예시적으로, 장치는 input layer, hidden layer, output layer를 포함할 수 있다. 장치를 구성하는 뉴럴 네트워크는 뉴런의 가중치를 학습을 통해 변화시킴으로써 임의의 입력(input)으로부터 예측하고자 하는 결과(output)를 추론할 수 있다.

[0033] 프로세서는 뉴럴 네트워크를 생성하거나, 뉴럴 네트워크를 훈련(train, 또는 학습(learn)하거나, 수신되는 입력데이터를 기초로 연산을 수행하고, 수행 결과를 기초로 정보 신호(information signal)를 생성하거나, 뉴럴 네트워크를 재훈련(retrain)할 수 있다. 뉴럴 네트워크의 모델들은 GoogleNet, AlexNet, VGG Network 등과 같은 CNN(Convolution Neural Network), R-CNN(Region with Convolution Neural Network), RPN(Region Proposal Network), RNN(Recurrent Neural Network), S-DNN(Stacking-based deep Neural Network), S-SDNN(State-Space Dynamic Neural Network), Deconvolution Network, DBN(Deep Belief Network), RBM(Restrcted Boltzman Machine), Fully Convolutional Network, LSTM(Long Short-Term Memory) Network, Classification Network 등 다양한 종류의 모델들을 포함할 수 있으나 이에 제한되지는 않는다. 프로세서는 뉴럴 네트워크의 모델들에 따른 연산을 수행하기 위한 하나 이상의 프로세서를 포함할 수 있다. 예를 들어 뉴럴 네트워크는 심층 뉴럴 네트워크(Deep Neural Network)를 포함할 수 있다.

[0034] 뉴럴 네트워크는 CNN(Convolutional Neural Network), RNN(Recurrent Neural Network), 퍼셉트론(perceptron), 다층 퍼셉트론(multilayer perceptron), FF(Feed Forward), RBF(Radial Basis Network), DFF(Deep Feed Forward), LSTM(Long Short Term Memory), GRU(Gated Recurrent Unit), AE(Auto Encoder), VAE(Variational Auto Encoder), DAE(Denoising Auto Encoder), SAE(Sparse Auto Encoder), MC(Markov Chain), HN(Hopfield Network), BM(Boltzmann Machine), RBM(Restricted Boltzmann Machine), DBN(Depp Belief Network), DCN(Deep Convolutional Network), DN(Deconvolutional Network), DCIGN(Deep Convolutional Inverse Graphics Network), GAN(Generative Adversarial Network), LSM(Liquid State Machine), ELM(Extreme Learning Machine), ESN(Echo State Network), DRN(Deep Residual Network), DNC(Differentiable Neural Computer), NTM(Neural Turning Machine), CN(Capsule Network), KN(Kohonen Network) 및 AN(Attention Network)를 포함할 수 있으나 이에 한정되는 것이 아닌 임의의 뉴럴 네트워크를 포함할 수 있음은 통상의 기술자가 이해할 것이다.

[0035] 본 발명의 예시적인 실시예에 따르면, 프로세서는 GoogleNet, AlexNet, VGG Network 등과 같은 CNN(Convolution Neural Network), R-CNN(Region with Convolution Neural Network), RPN(Region Proposal Network), RNN(Recurrent Neural Network), S-DNN(Stacking-based deep Neural Network), S-SDNN(State-Space Dynamic Neural Network), Deconvolution Network, DBN(Deep Belief Network), RBM(Restrcted Boltzman Machine), Fully Convolutional Network, LSTM(Long Short-Term Memory) Network, Classification Network, Generative Modeling, eXplainable AI, Continual AI, Representation Learning, AI for Material Design, 자연어 처리를 위한 BERT, SP-BERT, MRC/QA, Text Analysis, Dialog System, GPT-3, GPT-4, 비전 처리를 위한 Visual Analytics, Visual Understanding, Video Synthesis, ResNet 데이터 지능을 위한 Anomaly Detection, Prediction, Time-Series Forecasting, Optimization, Recommendation, Data Creation 등 다양한 인공지능 구조 및 알고리즘을 이용할 수 있으며, 이에 제한되지 않는다. 이하, 첨부된 도면을 참조하여 본 발명의 실시예를 상세하게 설명한다.

[0036] 도 1은 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치

(100)의 구성을 간략하게 도시한 블록도이며, 도 2는 서브그래프, 도 3은 학습 과정에서 업데이트되는 배치를 도시한 개념도이다. 도 4는 도 3과 비교되는 종래 그래프 뉴럴 네트워크의 메모리 관리 전자 장치의 학습 과정에서 업데이트되는 배치를 도시한 개념도이다.

[0037] 도 1 및 도 2에 도시된 바와 같이, 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치(100)는 학습 데이터셋을 배치 사이즈 단위의 서브그래프(200)로 분할하는 분할부(120), 각 서브그래프(200)에 포함된 노드를, 외부 서브그래프의 노드와 인접하게 배치된 경계 노드(220)와 상기 서브그래프 내부에 위치한 노드만 인접하게 배치된 내부 노드(210)로 분류하는 분류부(130), 상기 각 서브그래프(200)의 중간 임베딩(intermediate embedding)으로서 상기 경계 노드(220)에 대한 임베딩을 저장 및 관리하는 임베딩부(140) 및 상기 분할부(120), 상기 분류부(130) 및 상기 임베딩부(140)를 제어하는 적어도 하나의 프로세서(110)를 포함할 수 있다.

[0038] 또한, 기설정된 간격(staleness) 별로 상기 중간 임베딩을 업데이트하는 업데이트부(150)를 더 포함할 수 있다.

[0039] 상기 적어도 하나의 프로세서(110)는, 그래프 데이터셋을 각 서브그래프(200)로 분할하는 도중, 상기 각 서브그래프(200)에 포함된 노드를 경계 노드(220)와 내부 노드(210)로 분류하고, 상기 경계 노드(220)에 대한 중간 임베딩만을 저장하여 중간 임베딩 테이블로 관리하도록 구성될 수 있다.

[0040] 도 2를 참조하면, 그래프 데이터셋을 각 서브그래프(200)로 분할(Partition)으로 분할할 수 있다. 분할 #1인 서브그래프(200)에는 분할 #2, 분할 #3, ...인 서브그래프와 연결되지 않고 분할 #1인 서브그래프(200) 내부만으로 연결된 내부 노드(Inner nodes, 210) 및 분할 #2, 분할 #3, ...인 서브그래프와 연결된 노드인 경계 노드(Boundary nodes, 220)를 포함할 수 있다.

[0041] 그래프 뉴럴 네트워크 구조 및 특성을 이용하여 각 서브그래프(200)마다 이전 학습 단계에서 생성된 중간 임베딩들을 호스트 메모리에 저장하고, 추후 학습 단계에서 각 서브그래프(200)가 인접해 있던 경계 노드(220)만의 중간 임베딩을 포함한 배칭을 수행할 수 있다. 이를 통해 모델의 레이어 개수에 따라 지수적으로 증가하던 연산 및 메모리를 선형적으로 줄일 수 있다.

[0042] 따라서, 상기 적어도 하나의 프로세서(110)는, 상기 업데이트부(150)가 각 학습 단계에 참여한 서브그래프(200)와 인접한 외부 서브그래프의 중간 임베딩만을 업데이트하도록 구성될 수 있다.

[0043] 다만, 이 경우 모든 서브그래프(200)의 중간 임베딩을 호스트 메모리에 저장하여 메모리 부담이 있고, 중간 임베딩이 생성된 이전 학습 단계와 현재 학습 단계의 차가 일정(bounded staleness) 이상 커지면 오히려 학습 정확도에 악영향을 미칠 수 있다.

[0044] 따라서, 상기 적어도 하나의 프로세서(110)는, 상기 업데이트부(150)가 경계 노드(220)에 대해 상기 인접한 외부 서브그래프의 중간 임베딩이 업데이트된 이전 학습 단계를 확인하고, 상기 인접한 외부 서브그래프의 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초과한 서브그래프(200)만 상기 중간 임베딩을 업데이트하도록 구성될 수 있다.

[0045] 도 2를 참조하면, 상술한 과정을 통해 원 메모리(Original Memory)가 최적화 메모리(Optimized Memory)로 메모리를 최적화할 수 있다. 모델의 레이어 개수마다 경계 노드(220)에 대한 중간 임베딩만 호스트 메모리에 저장하여 불필요한 메모리를 제거하게 되고 연산 비용을 절감할 수 있다. 더하여, 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초과한 서브그래프(200)만을 업데이트하여 학습 모델의 정확도를 높게 유지할 수 있다.

[0046] 또한, 상기 적어도 하나의 프로세서(110)는, 각 학습 단계마다 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초과한 상기 인접한 외부 서브그래프를 가진 경계 노드(220)가 여러 개 존재할 경우, 각 학습 단계마다 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초과한 상기 인접한 외부 서브그래프의 개수가 최소인 서브그래프(200)를 배치로 선택하도록 구성될 수 있다.

[0047] 도 3을 참조하면, (a)는 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치(100)에 따른 배치 선택과 순전파 및 역전파 학습 과정을 나타낸 것이다. 이때 설정된 간격(staleness)을 2로 가정한다.

[0048] 도 4는 도 3의 (a)의 배치 선택 이후, 순전파 및 역전파 학습 과정을 실행하기 이전에 중간 임베딩을 업데이트하는 방안들을 나타낸 것이다. 도 4의 (b)는 원래 전체 임베딩을 기설정된 간격별로 순전파 연산을 통해 업데이트하는 종래 순전파 학습 방법이며, 도 4의 (c)는 기설정된 간격 별로 (a)에서 선택된 배치와없이 인접한 노드

의 임베딩만을 업데이트하는 종래 순전파 학습 방법이다. 도 4의 (d)는 (c)와 동일하게 기설정된 간격 별로 (a)에서 선택된 배치와 인접한 노드의 임베딩을 업데이트하지만, 이전 단계에서 업데이트한 노드들의 임베딩 중 현재 학습단계와 이전 단계의 차가 기설된 차이보다 큰 노드들만 업데이트하는 방법이다.

[0049] 도 4의 (b)는 모든 노드의 임베딩을 모두 업데이트하여야 하는 바, 학습 과정의 각 기설정된 단계마다 전체 노드에 대해 업데이트를 수행하여야 하고 메모리 및 업데이트 비용이 낭비될 수 있다.

[0050] 또한, 도 4의 (c)는 일부 노드의 임베딩만을 업데이트하기는 하나, 기설정된 간격 별로 선택된 배치와 인접한 모든 노드의 잦은 임베딩 업데이트가 발생할 수 있다.

[0051] 따라서, 도 4의 (d)와 같이, 본 발명의 일 실시예에 따른 전자 장치(100)는 인접한 서브그래프(200)에서 업데이트한 이전 학습 단계가 현재 학습 단계와 기설정된 차이가 날 경우에만 업데이트를 수행하여 메모리 및 업데이트 효율을 향상시킬 수 있다. 메모리의 연산 최적화 기술을 적용한 것이다.

[0052] 도 5는 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 전자 장치(100)의 메인 학습 모듈의 학습 과정을 도시한 프로세스도이다.

[0053] 상기 적어도 하나의 프로세서(110)는, 상기 배치(330)로 선택된 서브그래프의 인덱스(310)를 메인 학습 모듈(400)에 전달하고, 상기 메인 학습 모델(400)이 상기 서브그래프의 인덱스(310)와 상기 서브그래프에 인접한 외부 서브그래프의 중간 임베딩(320)을 로드하여 GPU에서 순전파 및 역전파 학습을 진행하기 위한 상기 배치(330)를 생성하도록 구성될 수 있다.

[0054] 도 5를 참조하면, 선택된 배치 인덱스(310)를 메인 학습 모듈(400)에 전달하면, 상기 배치 인덱스(310)에 대응하는 중간 임베딩(320)을 중간 임베딩 테이블에서 추출하여 매칭할 수 있다. 매칭된 인덱스(310) 및 중간 임베딩(320)를 이용하여 배치(330)를 생성하고 상기 배치(330)을 이용하여 학습을 수행할 수 있다.

[0055] 중간 임베딩 테이블은 각 서브그래프(200)의 중간 임베딩이 업데이트된 단계와 현재 학습 단계를 확인하여 탐욕 알고리즘을 통해 임베딩의 재업데이트가 요구되는 경계 노드의 개수가 최소인 서브그래프 인덱스를 메인 학습 모듈(400)에 전달할 수 있다.

[0056] 메인 학습 모듈(400)은 상기 서브그래프인 배치 인덱스(310)와 서브그래프와 인접한 서브그래프의 임베딩을 테이블에서 로드하여 배치(330)를 생성하고 GPU 장치에서 학습을 진행할 수 있다.

[0057] 한편, 본 발명의 일 실시예에 따른 전자 장치(100)에서 상기 중간 임베딩 테이블은 이전 임베딩 테이블보다 메모리 크기가 작을 수 있다.

[0058] 메모리 최적화되어 크기가 줄어든 중간 임베딩 테이블에서 임베딩을 로드 또는 저장할 때, 기존 임베딩 테이블과 상기 중간 임베딩 테이블의 다른 크기로 인해 각 서브그래프(200)의 오프셋이 수정되어 엠베딩 접근 중 메모리 오류가 발생할 수 있다.

[0059] 따라서, 상기 적어도 하나의 프로세서(110)는, 상기 이전 임베딩 테이블을 상기 중간 임베딩 테이블로 업데이트하는 경우, 상기 이전 임베딩 테이블의 상기 서브그래프 주소를 상기 중간 임베딩 테이블의 상기 서브그래프 주소로 리맵핑하도록 구성될 수 있다.

[0060] 이전 임베딩 테이블 내의 서브그래프(200)의 주소를 최적화된 중간 임베딩 테이블 내의 주소로 리맵핑(remapping)하여 메모리 오류 발생을 방지할 수 있다.

[0061] 한편, 본 발명의 일 실시예에 따른 서브그래프 샘플링 기반 그래프 뉴럴 네트워크의 메모리 관리 방법은 학습을 위한 그래프 데이터셋을 배치 사이즈 단위의 서브그래프(200)로 분할하는 단계(S510), 각 서브그래프에 포함된 노드를, 외부 서브그래프의 노드와 인접하게 배치된 경계 노드(220)와 상기 서브그래프 내부에 위치한 노드만 인접하게 배치된 내부 노드(210)로 분류하는 단계(S520) 및 상기 경계 노드(220)에 대한 중간 임베딩만을 상기 각 서브그래프(200)의 중간 임베딩 테이블로 저장 및 관리하는 단계(S530)를 포함할 수 있다.

[0062] 또한, 설정된 간격(staleness) 별로 상기 중간 임베딩을 업데이트하는 단계를 더 포함할 수 있다.

[0063] 상기 업데이트하는 단계는, 각 학습 단계에 참여한 서브그래프(200)와 인접한 외부 서브그래프의 중간 임베딩만을 업데이트하는 단계를 포함할 수 있다.

[0064] 상기 업데이트하는 단계는, 상기 인접한 외부 서브그래프의 중간 임베딩이 업데이트된 이전 학습 단계를 확인하는 단계 및 상기 인접한 외부 서브그래프의 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초

과한 서브그래프만 상기 중간 임베딩을 업데이트하는 단계를 더 포함할 수 있다.

[0065] 상기 업데이트하는 단계는, 각 학습 단계마다 현재 학습 단계와 상기 이전 학습 단계의 차가 기설정된 단계를 초과한 상기 인접한 외부 서브그래프의 개수가 최소인 서브그래프를 배치로 선택하는 단계를 더 포함할 수 있다.

[0066] 상기 배치로 선택된 서브그래프의 인덱스를 메인 학습 모듈에 전달하는 단계, 상기 메인 학습 모델이 상기 서브그래프의 인덱스와 상기 서브그래프에 인접한 외부 서브그래프의 중간 임베딩을 로드하여 상기 배치를 생성하는 단계 및 상기 배치를 이용하여 GPU에서 순전파 및 역전파 학습을 진행하는 단계를 더 포함할 수 있다.

[0067] 상기 중간 임베딩 테이블은 이전 임베딩 테이블보다 메모리 크기가 작을 수 있다.

[0068] 상기 이전 임베딩 테이블을 상기 중간 임베딩 테이블로 업데이트하는 경우, 상기 이전 임베딩 테이블의 상기 서브그래프 주소를 상기 중간 임베딩 테이블의 상기 서브그래프 주소로 리맵핑하는 단계를 더 포함할 수 있다. 상술한 내용과 중복되는 내용은 설명의 간략함을 위하여 생략하기로 한다.

[0069] 이를 통해 요구되는 호스트 메모리의 크기를 줄일 수 있으며 기설정된 간격을 고려하여 중간 임베딩의 업데이트를 수행하는 과정에서 업데이트시 요구되는 임베딩의 개수를 최소화하여 학습 시간을 최적화할 수 있다.

[0070] 한편, 개시된 실시예들은 컴퓨터에 의해 실행 가능한 명령어를 저장하는 기록매체의 형태로 구현될 수 있다. 명령어는 프로그램 코드의 형태로 저장될 수 있으며, 프로세서에 의해 실행되었을 때, 프로그램 모듈을 생성하여 개시된 실시예들의 동작을 수행할 수 있다. 기록매체는 컴퓨터로 읽을 수 있는 기록매체로 구현될 수 있다.

[0071] 컴퓨터가 읽을 수 있는 기록매체로는 컴퓨터에 의하여 해독될 수 있는 명령어가 저장된 모든 종류의 기록 매체를 포함한다. 예를 들어, ROM(Read Only Memory), RAM(Random Access Memory), 자기 테이프, 자기 디스크, 플래쉬 메모리, 광 데이터 저장장치 등이 있을 수 있다.

[0072] 이상에서와 같이 첨부된 도면을 참조하여 개시된 실시예들을 설명하였다. 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자는 본 발명의 기술적 사상이나 필수적인 특징을 변경하지 않고도, 개시된 실시예들과 다른 형태로 본 발명이 실시될 수 있음을 이해할 것이다. 개시된 실시예들은 예시적인 것이며, 한정적으로 해석되어서는 안 된다.

부호의 설명

[0073] 100: 전자 장치

110: 프로세서

120: 분할부

130: 분류부

140: 임베딩부

150: 업데이트부

200: 서브그래프

210: 내부 노드

220: 경계 노드

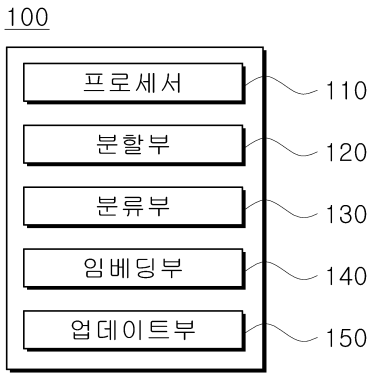
310: 배치 인덱스

320: 중간 임베딩

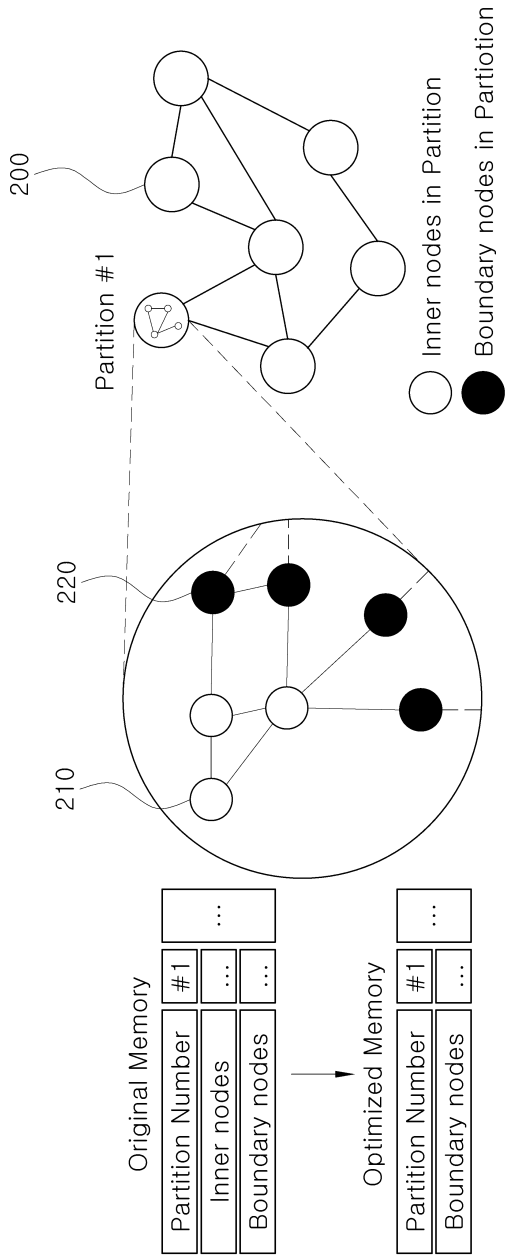
330: 배치

도면

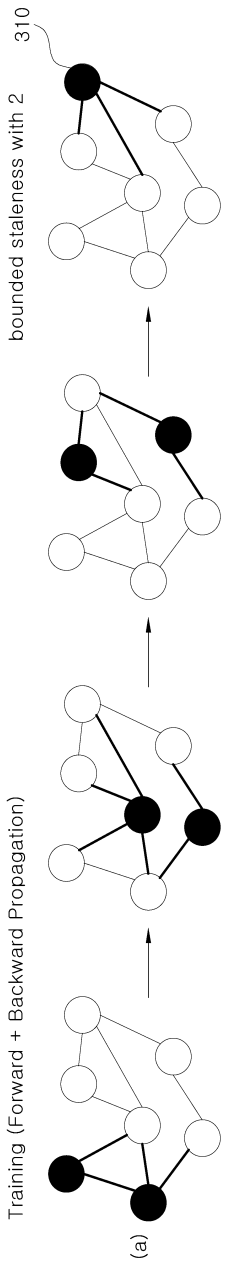
도면1



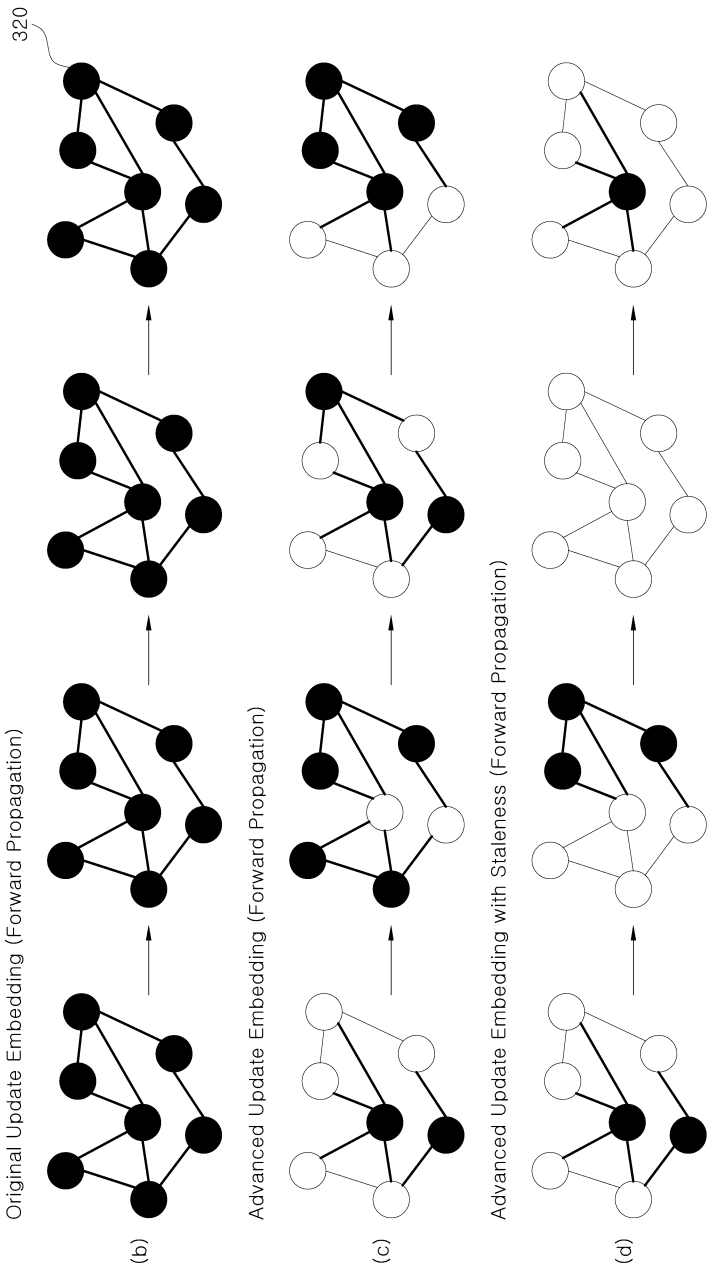
도면2


200
Partition #1
Inner nodes in Partition
Boundary nodes in Partiotion
220
210
Original Memory
Partition Number
Inner nodes
Boundary nodes
#1
Optimized Memory
Partition Number
Boundary nodes
#1

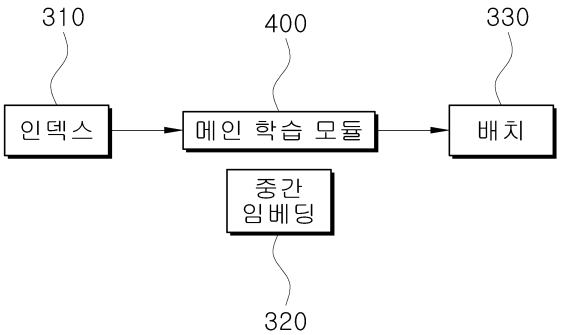
도면3


Training (Forward + Backward Propagation)
bounded staleness with 2
310
(a)

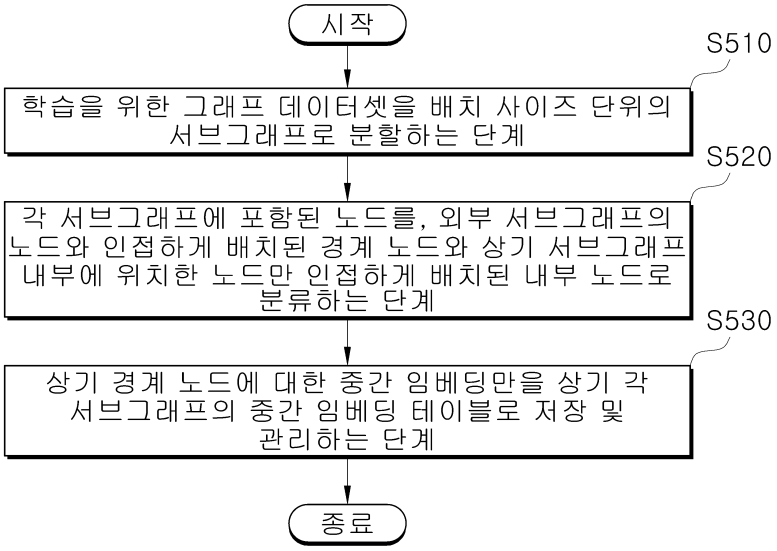
도면4


320
Original Update Embedding (Forward Propagation)
Advanced Update Embedding (Forward Propagation)
Advanced Update Embedding with Staleness (Forward Propagation)
(b)
(c)
(d)

도면5


310
400
330
인덱스
메인 학습 모듈
배치
중간
임베딩
320

도면6


시작
S510
학습을 위한 그래프 데이터셋을 배치 사이즈 단위의 서브그래프로 분할하는 단계
S520
각 서브그래프에 포함된 노드를, 외부 서브그래프의 노드와 인접하게 배치된 경계 노드와 상기 서브그래프 내부에 위치한 노드만 인접하게 배치된 내부 노드로 분류하는 단계
S530
상기 경계 노드에 대한 중간 임베딩만을 상기 각 서브그래프의 중간 임베딩 테이블로 저장 및 관리하는 단계
종료