



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0095438
(43) 공개일자 2025년06월26일

(51) 국제특허분류(Int. Cl.)
G06N 3/042 (2023.01) G06N 3/045 (2023.01)
G06N 3/063 (2023.01) G06N 3/08 (2023.01)
(52) CPC특허분류
G06N 3/042 (2023.01)
G06N 3/045 (2023.01)
(21) 출원번호 10-2024-0014410
(22) 출원일자 2024년01월30일
심사청구일자 2024년01월30일
(30) 우선권주장
1020230186397 2023년12월19일 대한민국(KR)

(71) 출원인
경희대학교 산학협력단
경기도 용인시 기흥구 덕영대로 1732 (서천동, 경희대학교 국제캠퍼스내)
(72) 발명자
이영구
경기도 수원시 영통구 청명북로 33, 431동 1704호
올라 이르판
경기도 수원시 영통구 매영로398번길 12, B03
(74) 대리인
두호특허법인

전체 청구항 수 : 총 15 항

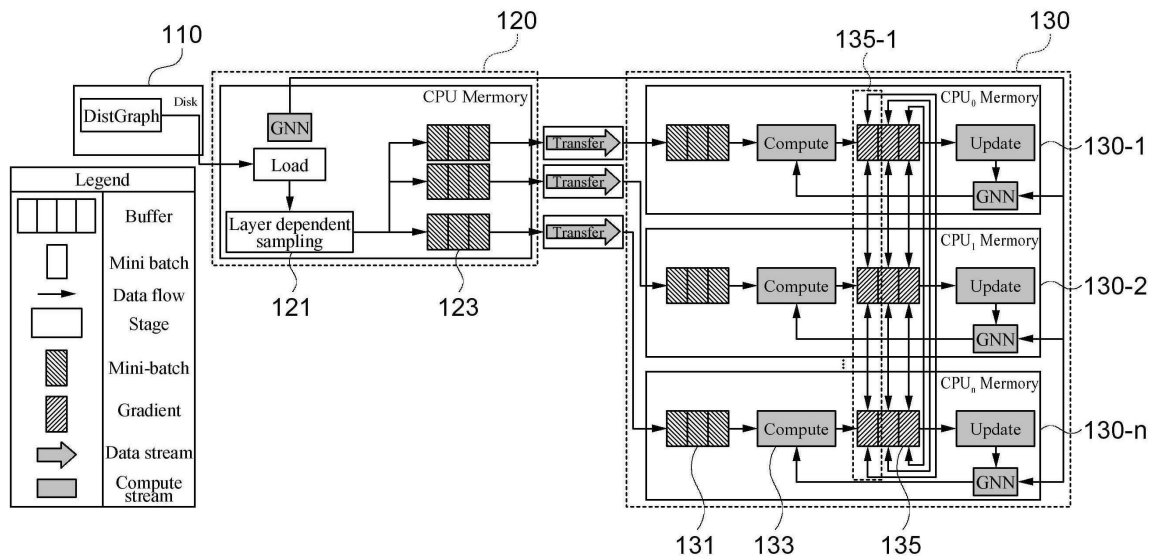
(54) 발명의 명칭 그래프 신경망 학습 지원 장치 및 방법

(57) 요약

다중 그래픽 처리 장치 환경에서 효율적인 학습을 위한 다중 큐와 레이어 의존 샘플링 구조를 가지는 그래프 신경망 학습 지원 장치 및 방법이 개시된다. 일 실시예에 따른 그래프 신경망 학습 지원 장치는 데이터 입출력을 위한 인터페이스부; 인터페이스부를 통해 수신한 데이터를 기초로 병렬 처리를 위한 미니배치를 생성하는 미니배치 생성부; 및 미니배치를 기초로 그래디언트(Gradient)를 연산하여 그래프 신경망을 업데이트하는 둘 이상의 신경망 학습부를 포함할 수 있다.

대표도

100



(52) CPC특허분류

G06N 3/063 (2013.01)

G06N 3/08 (2023.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711193818
과제번호	2021-0-00859-003
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	SW컴퓨팅산업원천기술개발
연구과제명	초거대 그래프의 지능적 고속 처리를 위한 그래프 DBMS 기술 개발
과제수행기관명	포항공과대학교 산학협력단
연구기간	2023.01.01 ~ 2023.12.31

명세서

청구범위

청구항 1

데이터 입출력을 위한 인터페이스부;

상기 인터페이스부를 통해 수신한 데이터를 기초로 병렬 처리를 위한 미니배치를 생성하는 미니배치 생성부; 및
상기 미니배치를 기초로 그래디언트(Gradient)를 연산하여 그래프 신경망을 업데이트하는 둘 이상의 신경망 학습부를 포함하는, 그래프 신경망 학습 지원 장치.

청구항 2

제 1 항에 있어서,

상기 미니배치 생성부는

상기 인터페이스부를 통해 수신한 데이터를 레이어 별로 샘플링하여 병렬 처리를 위한 둘 이상의 미니배치를 생성하는 중앙처리장치(central processing unit, CPU) 및 생성된 둘 이상의 미니배치를 각각 저장하는 병렬 연결된 둘 이상의 제 1 미니배치 버퍼를 포함하는, 그래프 신경망 학습 지원 장치.

청구항 3

제 2 항에 있어서,

상기 신경망 학습부는

상기 병렬 연결된 둘 이상의 제 1 미니배치 버퍼 중 어느 하나로부터 미니배치를 수신하는 제 2 미니배치 버퍼, 상기 제 2 미니배치 버퍼에 저장된 미니배치를 기초로 그래프 신경망을 학습시키기 위한 그래디언트를 연산하는 그래픽처리장치(Graphics Processing Unit, GPU) 및 상기 연산된 그래디언트를 저장하는 그래디언트 버퍼를 포함하는, 그래프 신경망 학습 지원 장치.

청구항 4

제 1 항에 있어서,

상기 신경망 학습부의 개수는 제 1 미니배치 버퍼의 개수와 동일한, 그래프 신경망 학습 지원 장치.

청구항 5

제 3 항에 있어서,

상기 둘 이상의 신경망 학습부는 병렬 연산을 위하여 서로 병렬로 연결된, 그래프 신경망 학습 지원 장치.

청구항 6

제 5 항에 있어서,

상기 신경망 학습부는

병렬로 연결된 다른 신경망 학습부의 그래디언트 버퍼를 확인하며,

둘 이상의 신경망 학습부 전체의 그래디언트 버퍼에 동일 레이어에 대한 그래디언트가 모두 저장된 경우, 그래디언트를 동기화하여 그래프 신경망을 업데이트하는, 그래프 신경망 학습 지원 장치.

청구항 7

제 6 항에 있어서,

상기 신경망 학습부에 포함된 각각의 그래픽처리장치는 다른 신경망 학습부의 그래픽처리장치의 연산 동작과 무관하게 비동기로 그래디언트를 연산하여 그래디언트 버퍼에 저장하는, 그래프 신경망 학습 지원 장치.

청구항 8

하나 이상의 프로세서들, 및

상기 하나 이상의 프로세서들에 의해 실행되는 하나 이상의 프로그램들을 저장하는 메모리를 구비한 컴퓨팅 장치에서 수행되는 방법으로서,

인터페이스를 통해 수신한 데이터를 기초로 병렬 처리를 위한 미니배치를 생성하는 단계; 및

상기 미니배치를 기초로 그래디언트(Gradient)를 연산하여 그래프 신경망을 업데이트하는 단계를 포함하는, 그래프 신경망 학습 지원 방법.

청구항 9

제 8 항에 있어서,

상기 미니배치를 생성하는 단계는

인터페이스를 통해 수신한 데이터를 레이어 별로 샘플링하여 병렬 처리를 위한 둘 이상의 미니배치를 생성하는 중앙처리장치(central processing unit, CPU) 및 생성된 둘 이상의 미니배치를 각각 저장하는 병렬 연결된 둘 이상의 제 1 미니배치 버퍼를 포함하는 미니배치 생성기를 이용하는, 그래프 신경망 학습 지원 방법.

청구항 10

제 9 항에 있어서,

상기 그래프 신경망을 업데이트하는 단계는

상기 병렬 연결된 둘 이상의 제 1 미니배치 버퍼 중 어느 하나로부터 미니배치를 수신하는 제 2 미니배치 버퍼, 상기 제 2 미니배치 버퍼에 저장된 미니배치를 기초로 그래프 신경망을 학습시키기 위한 그래디언트를 연산하는 그래픽처리장치(Graphics Processing Unit, GPU) 및 상기 연산된 그래디언트를 저장하는 그래디언트 버퍼를 포함하는 둘 이상의 신경망 학습기를 이용하는, 그래프 신경망 학습 지원 방법.

청구항 11

제 8 항에 있어서,

상기 신경망 학습기의 개수는 제 1 미니배치 버퍼의 개수와 동일한, 그래프 신경망 학습 지원 방법.

청구항 12

제 10 항에 있어서,

상기 둘 이상의 신경망 학습기는 병렬 연산을 위하여 서로 병렬로 연결된, 그래프 신경망 학습 지원 방법.

청구항 13

제 12 항에 있어서,

상기 그래프 신경망을 업데이트하는 단계는

병렬로 연결된 다른 신경망 학습부의 그래디언트 버퍼를 확인하며,

둘 이상의 신경망 학습부 전체의 그래디언트 버퍼에 동일 레이어에 대한 그래디언트가 모두 저장된 경우, 그래디언트를 동기화하여 그래프 신경망을 업데이트하는, 그래프 신경망 학습 지원 방법.

청구항 14

제 13 항에 있어서,

상기 신경망 학습기에 포함된 각각의 그래픽처리장치는 다른 신경망 학습기의 그래픽처리장치의 연산 동작과 무관하게 비동기로 그래디언트를 연산하여 그래디언트 버퍼에 저장하는, 그래프 신경망 학습 지원 방법.

청구항 15

비일시적 컴퓨터 판독 가능한 저장 매체(non-transitory computer readable storage medium)에 저장된 컴퓨터 프로그램으로서,

상기 컴퓨터 프로그램은 하나 이상의 명령어들을 포함하고, 상기 명령어들은 하나 이상의 프로세서들을 갖는 컴퓨팅 장치에 의해 실행될 때, 상기 컴퓨팅 장치로 하여금, 인터페이스를 통해 수신한 데이터를 기초로 병렬 처리를 위한 미니배치를 생성하는 단계; 및

상기 미니배치를 기초로 그래디언트를 연산하여 그래프 신경망을 업데이트하는 단계를 수행하도록 하는, 컴퓨터 프로그램.

발명의 설명

기술 분야

[0001] 다중 그래픽 처리 장치 환경에서 효율적인 학습을 위한 다중 큐와 레이어 의존 샘플링 구조를 가지는 그래프 신경망 학습 지원 장치 및 방법에 관한 것이다.

배경 기술

[0003] 소셜 네트워크, 유전자 표현 네트워크, 지식 그래프 등 현실세계의 다양한 데이터는 그래프의 형태를 가지고 있다. 이에 따라, 그래프 신경망(Graph Neural Networks, GNNs)은 머신러닝에서 중요한 분야로 성장하고 있다.

[0004] 그러나, 거대 그래프에 대한 자원 효율적인 GNN 학습은 주된 도전 분야이다. 애플리케이션에서 그래프는 보통 수억 개에서 수십억 개의 노드와 수백억 개에서 수천억 개의 에지를 포함하여 크기가 매우 크다. 나아가, 노드와 에지들은 GNN 입력으로 넣기 위한 특징 벡터와 연동되어 있다. 이에 따라, 단일 그래픽 처리 장치(GPU) 또는 단일 머신에서 거대 그래프를 학습하는 것은 제한된 자원이라는 점에서 매우 어려운 일이다.

선행기술문헌

특허문헌

[0006] (특허문헌 0001) 대한민국 공개특허 제 10-2023-0170397 호

발명의 내용

해결하려는 과제

[0007] 다중 그래픽 처리 장치 환경에서 효율적인 학습을 위한 다중 큐와 레이어 의존 샘플링 구조를 가지는 그래프 신경망 학습 지원 장치 및 방법을 제공하는데 목적이 있다.

과제의 해결 수단

[0009] 일 양상에 따르면, 그래프 신경망 학습 지원 장치는 데이터 입출력을 위한 인터페이스부; 인터페이스부를 통해 수신한 데이터를 기초로 병렬 처리를 위한 미니배치를 생성하는 미니배치 생성부; 및 미니배치를 기초로 그래디언트(Gradient)를 연산하여 그래프 신경망을 업데이트하는 둘 이상의 신경망 학습부를 포함할 수 있다.

[0010] 미니배치 생성부는 인터페이스부를 통해 수신한 데이터를 레이어 별로 샘플링하여 병렬 처리를 위한 둘 이상의 미니배치를 생성하는 중앙처리장치(central processing unit, CPU) 및 생성된 둘 이상의 미니배치를 각각 저장하는 병렬 연결된 둘 이상의 제 1 미니배치 버퍼를 포함할 수 있다.

[0011] 신경망 학습부는 병렬 연결된 둘 이상의 제 1 미니배치 버퍼 중 어느 하나로부터 미니배치를 수신하는 제 2 미니배치 버퍼, 제 2 미니배치 버퍼에 저장된 미니배치를 기초로 그래프 신경망을 학습시키기 위한 그래디언트를 연산하는 그래픽처리장치(Graphics Processing Unit, GPU) 및 연산된 그래디언트를 저장하는 그래디언트 버퍼를 포함할 수 있다.

[0012] 신경망 학습부의 개수는 제 1 미니배치 버퍼의 개수와 동일할 수 있다.

[0013] 둘 이상의 신경망 학습부는 병렬 연산을 위하여 서로 병렬로 연결될 수 있다.

[0014] 신경망 학습부는 병렬로 연결된 다른 신경망 학습부의 그래디언트 버퍼를 확인하며, 둘 이상의 신경망 학습부 전체의 그래디언트 버퍼에 동일 레이어에 대한 그래디언트가 모두 저장된 경우, 그래디언트를 동기화하여 그래프 신경망을 업데이트할 수 있다.

[0015] 신경망 학습부에 포함된 각각의 그래픽처리장치는 다른 신경망 학습부의 그래픽처리장치의 연산 동작과 무관하게 비동기로 그래디언트를 연산하여 그래디언트 버퍼에 저장할 수 있다.

[0016] 일 양상에 따르면, 하나 이상의 프로세서들, 및 하나 이상의 프로세서들에 의해 실행되는 하나 이상의 프로그램들을 저장하는 메모리를 구비한 컴퓨팅 장치에서 수행되는 방법은 인터페이스를 통해 수신한 데이터를 기초로 병렬 처리를 위한 미니배치를 생성하는 단계; 및 미니배치를 기초로 그래디언트(Gradient)를 연산하여 그래프 신경망을 업데이트하는 단계를 포함할 수 있다.

[0017] 미니배치를 생성하는 단계는 인터페이스를 통해 수신한 데이터를 레이어 별로 샘플링하여 병렬 처리를 위한 둘 이상의 미니배치를 생성하는 중앙처리장치(central processing unit, CPU) 및 생성된 둘 이상의 미니배치를 각각 저장하는 병렬 연결된 둘 이상의 제 1 미니배치 버퍼를 포함하는 미니배치 생성기를 이용할 수 있다.

[0018] 그래프 신경망을 업데이트하는 단계는 병렬 연결된 둘 이상의 제 1 미니배치 버퍼 중 어느 하나로부터 미니배치를 수신하는 제 2 미니배치 버퍼, 제 2 미니배치 버퍼에 저장된 미니배치를 기초로 그래프 신경망을 학습시키기 위한 그래디언트를 연산하는 그래픽처리장치(Graphics Processing Unit, GPU) 및 연산된 그래디언트를 저장하는 그래디언트 버퍼를 포함하는 둘 이상의 신경망 학습기를 이용할 수 있다.

[0019] 그래프 신경망을 업데이트하는 단계는 병렬로 연결된 다른 신경망 학습부의 그래디언트 버퍼를 확인하며, 둘 이상의 신경망 학습부 전체의 그래디언트 버퍼에 동일 레이어에 대한 그래디언트가 모두 저장된 경우, 그래디언트를 동기화하여 그래프 신경망을 업데이트할 수 있다.

[0020] 신경망 학습기에 포함된 각각의 그래픽처리장치는 다른 신경망 학습기의 그래픽처리장치의 연산 동작과 무관하게 비동기로 그래디언트를 연산하여 그래디언트 버퍼에 저장할 수 있다.

[0021] 일 양상에 따르면, 비일시적 컴퓨터 판독 가능한 저장 매체(non-transitory computer readable storage

medium)에 저장된 컴퓨터 프로그램으로서, 컴퓨터 프로그램은 하나 이상의 명령어들을 포함하고, 명령어들은 하나 이상의 프로세서들을 갖는 컴퓨팅 장치에 의해 실행될 때, 컴퓨팅 장치로 하여금, 인터페이스를 통해 수신한 데이터를 기초로 병렬 처리를 위한 미니배치를 생성하는 단계; 및 미니배치를 기초로 그래디언트를 연산하여 그래프 신경망을 업데이트하는 단계를 수행하도록 할 수 있다.

발명의 효과

[0023] 미니배치 버퍼 및 그래디언트 버퍼를 이용하여 병렬로 연결된 GPU의 연산 효율을 증대시킬 수 있다.

도면의 간단한 설명

[0025] 도 1은 일 실시예에 그래프 신경망 학습 지원 장치의 구성도이다.

도 2는 일 실시예에 따른 그래프 신경망 학습 지원 장치의 구성을 설명하기 위한 예시도이다.

도 3은 일 실시예에 따른 그래프 신경망 학습 지원 방법을 도시한 흐름도이다.

도 4는 예시적인 실시예들에서 사용되기에 적합한 컴퓨팅 장치를 포함하는 컴퓨팅 환경을 예시하여 설명하기 위한 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0026] 이하, 도면을 참조하여 본 발명의 구체적인 실시형태를 설명하기로 한다. 이하의 상세한 설명은 본 명세서에서 기술된 방법, 장치 및/또는 시스템에 대한 포괄적인 이해를 돕기 위해 제공된다. 그러나 이는 예시에 불과하며 본 발명은 이에 제한되지 않는다.

[0027] 본 발명의 실시예들을 설명함에 있어서, 본 발명과 관련된 공지기술에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명을 생략하기로 한다. 그리고, 후술되는 용어들은 본 발명에서의 기능을 고려하여 정의된 용어들로서 이는 사용자, 운용자의 의도 또는 관례 등에 따라 달라질 수 있다. 그러므로 그 정의는 본 명세서 전반에 걸친 내용을 토대로 내려져야 할 것이다. 상세한 설명에서 사용되는 용어는 단지 본 발명의 실시예들을 기술하기 위한 것이며, 결코 제한적이어서는 안 된다. 명확하게 달리 사용되지 않는 한, 단수 형태의 표현은 복수 형태의 의미를 포함한다. 본 설명에서, "포함" 또는 "구비"와 같은 표현은 어떤 특성들, 숫자들, 단계들, 동작들, 요소들, 이들의 일부 또는 조합을 가리키기 위한 것이며, 기술된 것 이외에 하나 또는 그 이상의 다른 특성, 숫자, 단계, 동작, 요소, 이들의 일부 또는 조합의 존재 또는 가능성을 배제하도록 해석되어서는 안 된다.

[0028] 또한, 제1, 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 구성 요소들은 용어들에 의해 한정되어서는 안 된다. 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로 사용될 수 있다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제1 구성 요소는 제2 구성 요소로 명명될 수 있고, 유사하게 제2 구성 요소도 제1 구성 요소로 명명될 수 있다.

[0029] 도 1은 일 실시예에 그래프 신경망 학습 지원 장치의 구성도이다.

[0030] 도 1을 참조하면, 그래프 신경망 학습 지원 장치(100)는 데이터 입출력을 위한 인터페이스부(110), 인터페이스부(110)를 통해 수신한 데이터를 기초로 병렬 처리를 위한 미니배치를 생성하는 미니배치 생성부(120) 및 미니배치를 기초로 그래디언트(Gradient)를 연산하여 그래프 신경망을 업데이트하는 둘 이상의 신경망 학습부(130)를 포함할 수 있다.

[0031] 일 예로, 그래프 신경망 학습 지원 장치(100)는 그래프 신경망(Graph Neural Network, GNN) 위한 학습 파이프라인 구조인 MQ-LAS(Multi-Queued and LAYer-dependent Sampling based)를 가질 수 있다. 그래프 신경망 학습 지원 장치(100)는 데이터 접근, 전송 및 계산을 인터리브(interleave)하고 레이어 의존 샘플링(layer-dependent sampling)을 활용하여 GPU 활용률을 높일 수 있다. 이에 따라, 그래프 신경망 학습 지원 장치(100)는 기존 시스템과 비교하여 보다 정확하고 빠르게 데이터 연산을 수행할 수 있다.

[0032] 일 예로, 그래프 신경망 학습 지원 장치(100)는 학습 시간을 단축하기 위하여 비동기 학습을 수행할 수 있다. 그래프 신경망 학습 지원 장치(100)는 CPU와 GPU 모두에서 미니배치를 유지하기 위한 비동기 데이터 이동과 그

래프 신경망 모델의 일관성을 유지하기 위한 GPU상에서만 그라디언트를 유지하는 파이프라인으로 구성될 수 있다. 여기서, 미니배치는 데이터 버퍼에 버퍼링되어 미니배치 생성 시간엔 감춰지지만 학습 파이프라인에서는 사용될 수 있다. 또한 다중 GPU에서 비동기 그라디언트 축적을 수행하며 그래프 신경망 모델을 업데이트할 수 있다. 특히, 그래프 신경망 학습 지원 장치(100)는 그라디언트를 그라디언트 버퍼에 축적하고 각각의 GPU에서 특정 시점의 미니배치에 대한 그라디언트 축적이 끝나는 대로 그래프 신경망 모델을 업데이트할 수 있다. 즉, GPU에 대한 그라디언트 축적은 각각의 GPU에서 비동기적으로 수행하나, 그래프 신경망 모델의 업데이트는 각각의 GPU에서 특정 시점 또는 특정 레이어에 대한 미니배치의 그라디언트 연산이 완료되어 버퍼에 버퍼링되는 시점에 수행됨으로써 동기를 맞출 수 있다.

[0033] 도 2는 일 실시예에 따른 그래프 신경망 학습 지원 장치의 구성을 설명하기 위한 예시도이다.

[0034] 일 실시예에 따르면, 미니배치 생성부(120)는 인터페이스부(110)를 통해 수신한 데이터를 레이어 별로 샘플링하여 병렬 처리를 위한 둘 이상의 미니배치를 생성하는 중앙처리장치(central processing unit, CPU)(121) 및 생성된 둘 이상의 미니배치를 각각 저장하는 병렬 연결된 둘 이상의 제 1 미니배치 버퍼(123)를 포함할 수 있다.

[0035] 일 예로, 그래프 $G = (V, E)$ 가 있을 때, $V = (v_1, v_2, \dots, v_N)$ 는 노드들의 집합이고 $E = (e_{ij} \mid v_i, v_j \in V)$ 는 노드 v_i 과 v_j 를 잇는 에지(edge)들의 집합일 수 있다.

[0036] 미니배치 생성부(120)는 연속적으로 그리고 동시에 미니배치를 생성되어 제 1 미니배치 버퍼(123)에 삽입할 수 있다. 이렇게 생성된 미니배치들은 제 1 미니배치 버퍼(123)를 통해 신경망 학습부(130)로 전송될 수 있다. 미니배치가 신경망 학습부(130)에 도착하면, 미니배치는 그래프 신경망 모델 훈련을 위해 추출될 수 있다. 예를 들어, 신경망 학습부(130)는 지속적으로 유입되는 새로운 미니배치의 대기열을 계속 사용함으로써 GPU 활용을 최적화 할 수 있다. 이에 따라, 다른 신경망 학습부의 GPU와 중간에 그라디언트 동기화를 고려하지 않고 다음 미니배치로의 끊임없는 그라디언트 연산을 수행할 수 있다. 이후, 모든 그라디언트가 준비되면 축적이 발생하고, 모든 GPU에서 모델 업데이트를 수행할 수 있다.

[0037] 일 예에 따르면, 미니배치 생성부(120)는 데이터셋을 수신하며, 미니배치들을 생성하여 제 1 미니배치 버퍼(123)에 저장할 수 있다. 특정 레이어에서 다음 레이어로 샘플링 되는 노드들은 이웃들의 합집합으로 아래와 같이 표시될 수 있다.

[0038] [수학식 1]

$$V^{l-1} = \bigcup_{v_i \in S_l} N(v_i)$$

[0039]

[0041] 여기서, V^{l-1} 는 레이어 l-1에서 샘플링 된 노드이며, 노드 v_i 의 이웃 노드의 집합이다. S_l 는 l번째 층에서 샘플링한 노드의 집합이며, $N(v_i)$ 는 v_i 의 이웃의 집합이다. 샘플링 과정 동안 확률 할당은 $\{p_i^{l-1}\}_{v_i \in V^{l-1}}$ 에 의해 지정된 노드들인 V^{l-1} 로 제한된다. 특히, 중요한 확률은 다음과 같이 정의될 수 있다.

[0042] [수학식 2]

$$p_i^{l-1} = \frac{\|R^l L_{v,i}\|_2^2}{\|R^l L\|_F^2}$$

[0043]

[0045] 여기서, l번째 및 l-1번째 레이어에서 샘플링된 노드의 집합이 결정될 수 있다. 그리고 각 노드 v_i 에 확률 $p_1^{l-1} \dots p_{|v|}^{l-1}$ 가 배정될 수 있다. 그 대각 행렬 $S_{s,s}^{l-1}$ 는 아래와 같이 정의될 수 있다.

[0046] [수학식 3]

$$S_{s,s}^{i-1} = \begin{cases} \frac{1}{s_{i-1} p_{i_k^{i-1}}^{i-1}} & s = i_k^{i-1} \\ 0 & otherwise \end{cases}$$

[0049] 여기서, s_{i-1} 는 레이어 1-1에서 샘플링 된 노드 숫자이다.

[0050] 일 예로, 미니배치 생성부(120)는 1번째 레이어에서의 샘플들을 정의하는 동안 중간 임베딩 또는 활성화 행렬에 대한 정보가 없기 때문에 기본적인 샘플링 방식을 사용하며, 이는 행 선택 행렬 R^1 과 정규화된 라플라시안 행렬 L 에 의존하는 것일 수 있다. 여기서, 행 선택 행렬 R^1 는 다음과 같이 정의될 수 있다.

[0051] [수학식 4]

$$Q_{c,r}^i = \begin{cases} 1 & (c,r) = (c, r_k^i) \\ 0 & otherwise \end{cases}$$

[0054] 여기서, $\{i_k^i\}$ 와 $\{i_k^{i-1}\}$ 는 노드집합 S_i 과 S_{i-1} 를 의미한다. 라플라시안 행렬 L 는 다음과 같이 정의될 수 있다.

[0055] [수학식 5]

$$L = \hat{D}^{-1/2} \hat{A} \hat{D}^{-1/2}$$

[0058] 이때, $\hat{A} = A + I$ 에서 A 는 인접행렬이며, $\hat{D}$ 는 $\hat{D}_{i,i} = \sum_j \hat{A}_{i,j}$ 를 만족하는 대각행렬이다. 샘플링된 미니배치는 라플라시안 인접행렬과 따라오는 임베딩 (또는, 레이어 0의 특징값) $\frac{1}{p_k^{i-1}} \hat{L}_{*,k}$ 와 $Z_{k,*}^{i-1}$ 로 나타낼 수 있다. $\tilde{L}^i \leftarrow D_{\tilde{p}^i}^{-1} \tilde{p}^i$ 는 그래디언트 소실이나 폭발을 방지하기 위한 정규화된 라플라시안 행렬이다.

[0059] 일 실시예에 따르면, 신경망 학습부(130)는 병렬 연결된 둘 이상의 제 1 미니배치 버퍼(123) 중 어느 하나로부터 미니배치를 수신하는 제 2 미니배치 버퍼(131), 제 2 미니배치 버퍼에 저장된 미니배치를 기초로 그래프 신경망을 학습시키기 위한 그래디언트를 연산하는 그래픽처리장치(Graphics Processing Unit, GPU)(133) 및 연산된 그래디언트를 저장하는 그래디언트 버퍼(135)를 포함할 수 있다.

[0060] 일 예로, 신경망 학습부(130)의 개수는 제 1 미니배치 버퍼(123)의 개수와 동일할 수 있다. 예를 들어, 도 2에서와 같이 제 1 미니배치 버퍼(123)는 각각 하나의 신경망 학습부(130-1, ..., 130-n)와 연결될 수 있다. 이때, 신경망 학습부(130-1, ..., 130-n)는 병렬 연산을 위하여 서로 병렬로 연결될 수 있다.

[0061] 일 예에 따르면, 생성된 미니배치들은 신경망 학습부(130)의 제 2 미니배치 버퍼(131)로 전송된다. 이를 통해, 데이터 전송 매커니즘을 최적화하여 대기 시간을 최소화하고 GPU가 미니배치에 빠르고 효율적으로 접근할 수 있다.

[0062] 일 예에 따르면, 신경망 학습부(130)의 GPU(133)는 입력된 미니배치를 연산하여 그래디언트를 생성할 수 있다. 미니배치를 생성하고 대기열에 추가한 후 다음과 같이 그래프 신경망을 학습하는데 미니배치를 사용할 수 있다.

[0063] [수학식 6]

$$Z^{i+1}(v) = GNN^i([Z_{agg}^i(v), Z^i(v)]; W^i)$$

$$Z_{agg}^i(v) = Mean(H^i(v))$$

[0067] 여기서 $H^l(v)$ 는 1번째 레이어에서 샘플링된 노드 v 의 이웃들의 특징 집합이다. $H^{l+1}(v)$ 는 v 의 $(l+1)$ 번째 레이어에서의 임베딩 값이다. $[h_{agg}^l(v), h^l(v)]$ 는 neighbor aggregation 방식으로 쓰이는 GraphSAGE모형을 위해 이웃의 정보를 축적한 후 레이어 l 에서의 노드 v 의 임베딩값을 값을 연결한 것이다. GCN 모델에서는 위의 $Z^{l+1}(v)$ 의 이웃 축적은 아래와 같이 대체될 수 있다.

[0068] [수학식 7]

[0069]
$$Z^{l+1}(v) = GNN^l(Z_{agg}^l(v); W^l)$$

[0071] 일 예에 따르면, 복수개의 신경망 학습부(130-1, ..., 130-n)들은 각각 다른 신경망 학습부와 즉시 그래디언트를 동기화하는 대신 그래디언트를 그래디언트 버퍼(135)에 저장할 수 있다. 이를 통해, 그래프 신경망 훈련 과정이 다음 데이터 배치로 원활하게 이어질 수 있게 하여 GPU의 활용도를 높일 수 있다. 즉, 다른 신경망 학습부의 GPU가 연산하는 동안 기다리지 않고도 다음 단계의 레이어에 해당하는 미니배치의 연산으로 넘어가게 하여 훈련 과정 동안 계산 효율성이 손실되지 않도록 보장할 수 있다.

[0072] 일 예에 따르면, 신경망 학습부의 GPU는 순전파를 수행하며, 이때 손실값 $loss_s$ 와 그에 대응하는 그래디언트 ∇_l^s 는 다음과 같이 계산되어 그래디언트 버퍼(135)로 입력될 수 있다.

[0073] [수학식 8]

[0074]
$$loss_s = - \sum_{v \in N_s(v)} y_v \log(pr_v) + (1 - y_v) \log((1 - pr_v))$$

[0075]
$$\nabla_l^s = W^l loss_s$$

[0077] 여기서, y 는 실제 라벨 값이며 pr 은 예측값을 의미한다.

[0078] 일 실시예에 따르면, 신경망 학습부(130)는 병렬로 연결된 다른 신경망 학습부의 그래디언트 버퍼를 확인하며, 둘 이상의 신경망 학습부 전체의 그래디언트 버퍼에 동일 레이어에 대한 그래디언트가 모두 저장된 경우, 그래디언트를 동기화하여 그래프 신경망을 업데이트할 수 있다. 이때, 신경망 학습부에 포함된 각각의 그래픽처리장치는 다른 신경망 학습부의 그래픽처리장치의 연산 동작과 무관하게 비동기로 그래디언트를 연산하여 그래디언트 버퍼에 저장할 수 있다.

[0079] 도 3를 참조하면, 신경망 학습부(130)의 각각의 GPU는 그래디언트를 연산하여 그래디언트 버퍼(135)에 저장할 수 있다. 이때, 각각의 GPU에서 연산되는 속도 또는 시간이 다를 수 있는 바, 그래디언트 버퍼로 저장되는 그래디언트의 동기가 맞지 않을 수 있다. 예를 들어, GPU가 특정 레이어의 미니배치를 연산한 그래디언트를 그래디언트 버퍼에 순차적으로 입력할 수 있다. 예를 들어, 특정 레이어의 미니배치를 연산한 그래디언트가 그래디언트 버퍼(135-1)로 입력될 수 있다. 이때, 신경망 학습부(130-2)의 연산이 늦어지는 경우, 신경망 학습부(130-2)의 그래디언트 버퍼(135-1)에는 특정 레이어의 미니배치를 연산한 그래디언트가 입력되지 않을 수 있다. 이러한 경우, 다른 신경망 학습부는 그래프 신경망 모델의 학습을 하지 않고 대기할 수 있다. 반면, 다른 신경망 학습부는 신경망 학습부(130-2)의 연산 지연과 무관하게 다음 단계의 미니배치를 연산하여 그래디언트를 구할 수 있다.

[0080] 일 예에 따르면, 신경망 학습부는 GPU간의 그래디언트 동기화와 축적을 수행할 수 있다. 예를 들어, 신경망 학습부는 다른 신경망 학습부의 GPU에서 얻은 그래디언트를 동기화할 수 있다. 동기화된 그래디언트 업데이트는 모든 GPU에서 그래디언트 연산이 종료된 후 동기화되어 수행될 수 있다. 즉, 그래디언트 연산은 비동기적으로 수행되어 그래디언트 버퍼에 저장되나 그래디언트 업데이트는 동기화되어 수행될 수 있다.

[0081] 일 예로, 그래디언트의 동기화를 위해 신경망 학습부는 모든 그래디언트를 아래와 같이 축적할 수 있다.

[0082] [수학식 9]

$$[0083] \quad \nabla_l^{acc} = \nabla_l^{d1} + \nabla_l^{d2} + \nabla_l^{d3} \dots + \nabla_l^{dn}$$

[0085] 여기서, ∇_l^{acc} 는 레이어 1에서의 축적된 그래디언트이며, ∇_1^{d1} 는 소정 미니배치의 GPU d₁에서의 그래디언트이다.

[0086] 일 예에 따르면, 신경망 학습부는 누적된 그래디언트에 기반하여 각 GPU에서 그래프 신경망 모델을 업데이트할 수 있다. 일 예로, 신경망 학습부는 아래 수식과 같이 그래프 신경망 모델을 업데이트할 수 있다.

[0087] [수학식 10]

$$[0088] \quad W^l = W^l - \eta \nabla_l^{acc}$$

[0090] 여기서, W^l 은 레이어 1에서의 모델의 가중치이며 η 는 학습률이다.

[0091] 도 3은 일 실시예에 따른 그래프 신경망 학습 지원 방법을 도시한 흐름도이다.

[0092] 일 실시예에 따르면, 그래프 신경망 학습 지원 장치는 하나 이상의 프로세서들 및 하나 이상의 프로세서들에 의해 실행되는 하나 이상의 프로그램들을 저장하는 메모리를 구비한 컴퓨팅 장치일 수 있다.

[0093] 일 실시예에 따르면, 그래프 신경망 학습 지원 장치는 인터페이스를 통해 수신한 데이터를 기초로 병렬 처리를 위한 미니배치를 생성할 수 있으며(310), 미니배치를 기초로 그래디언트(Gradient)를 연산하여 그래프 신경망을 업데이트할 수 있다(320).

[0094] 일 예에 따르면, 그래프 신경망 학습 지원 장치는 미니배치를 생성하는 단계에서 인터페이스를 통해 수신한 데이터를 레이어 별로 샘플링하여 병렬 처리를 위한 둘 이상의 미니배치를 생성하는 중앙처리장치(central processing unit, CPU) 및 생성된 둘 이상의 미니배치를 각각 저장하는 병렬 연결된 둘 이상의 제 1 미니배치 버퍼를 포함하는 미니배치 생성기를 이용할 수 있다. 또한, 그래프 신경망 학습 지원 장치는 그래프 신경망을 업데이트하는 단계에서 병렬 연결된 둘 이상의 제 1 미니배치 버퍼 중 어느 하나로부터 미니배치를 수신하는 제 2 미니배치 버퍼, 제 2 미니배치 버퍼에 저장된 미니배치를 기초로 그래프 신경망을 학습시키기 위한 그래디언트를 연산하는 그래픽처리장치(Graphics Processing Unit, GPU) 및 연산된 그래디언트를 저장하는 그래디언트 버퍼를 포함하는 둘 이상의 신경망 학습기를 이용할 수 있다.

[0095] 일 실시예에 따르면, 그래프 신경망 학습 지원 장치는 그래프 신경망을 업데이트하는 단계에서 병렬로 연결된 다른 신경망 학습부의 그래디언트 버퍼를 확인하며, 둘 이상의 신경망 학습부 전체의 그래디언트 버퍼에 동일 레이어에 대한 그래디언트가 모두 저장된 경우, 그래디언트를 동기화하여 그래프 신경망을 업데이트할 수 있다. 이때, 신경망 학습기에 포함된 각각의 그래픽처리장치는 다른 신경망 학습기의 그래픽처리장치의 연산 동작과 무관하게 비동기로 그래디언트를 연산하여 그래디언트 버퍼에 저장할 수 있다.

[0096] 도 3의 실시예 중 도 1 및 도 2를 참조하여 설명한 내용과 중복되는 실시예는 생략하였다.

[0097] 도 4는 예시적인 실시예들에서 사용되기에 적합한 컴퓨팅 장치를 포함하는 컴퓨팅 환경(10)을 예시하여 설명하기 위한 블록도이다. 도시된 실시예에서, 각 컴포넌트들은 이하에 기술된 것 이외에 상이한 기능 및 능력을 가질 수 있고, 이하에 기술된 것 이외에도 추가적인 컴포넌트를 포함할 수 있다.

[0098] 도시된 컴퓨팅 환경(10)은 컴퓨팅 장치(12)를 포함한다. 일 실시예에서, 컴퓨팅 장치(12)는 비디오 표현 학습 장치일 수 있다.

[0099] 컴퓨팅 장치(12)는 적어도 하나의 프로세서(14), 컴퓨터 판독 가능 저장 매체(16) 및 통신 버스(18)를 포함한다. 프로세서(14)는 컴퓨팅 장치(12)로 하여금 앞서 언급된 예시적인 실시예에 따라 동작하도록 할 수 있다. 예컨대, 프로세서(14)는 컴퓨터 판독 가능 저장 매체(16)에 저장된 하나 이상의 프로그램들을 실행할 수 있다. 하나 이상의 프로그램들은 하나 이상의 컴퓨터 실행 가능 명령어를 포함할 수 있으며, 컴퓨터 실행 가능 명령어는 프로세서(14)에 의해 실행되는 경우 컴퓨팅 장치(12)로 하여금 예시적인 실시예에 따른 동작들을 수행하도록 구성될 수 있다.

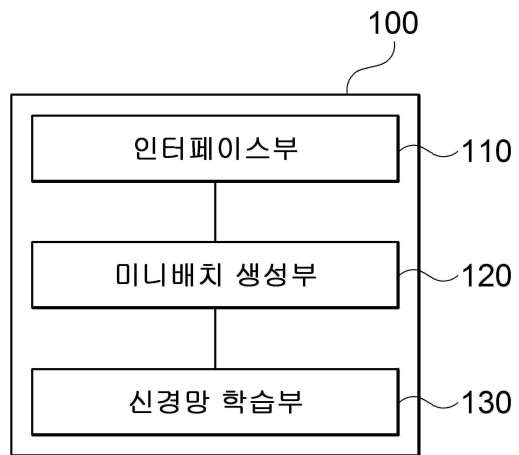
- [0100] 컴퓨터 판독 가능 저장 매체(16)는 컴퓨터 실행 가능 명령어 내지 프로그램 코드, 프로그램 데이터 및/또는 다른 적합한 형태의 정보를 저장하도록 구성된다. 컴퓨터 판독 가능 저장 매체(16)에 저장된 프로그램(20)은 프로세서(14)에 의해 실행 가능한 명령어의 집합을 포함한다. 일 실시예에서, 컴퓨터 판독 가능 저장 매체(16)는 메모리(랜덤 액세스 메모리와 같은 휘발성 메모리, 비휘발성 메모리, 또는 이들의 적절한 조합), 하나 이상의 자기 디스크 저장 디바이스들, 광학 디스크 저장 디바이스들, 플래시 메모리 디바이스들, 그 밖에 컴퓨팅 장치(12)에 의해 액세스되고 원하는 정보를 저장할 수 있는 다른 형태의 저장 매체, 또는 이들의 적합한 조합일 수 있다.
- [0101] 통신 버스(18)는 프로세서(14), 컴퓨터 판독 가능 저장 매체(16)를 포함하여 컴퓨팅 장치(12)의 다른 다양한 컴포넌트들을 상호 연결한다.
- [0102] 컴퓨팅 장치(12)는 또한 하나 이상의 입출력 장치(24)를 위한 인터페이스를 제공하는 하나 이상의 입출력 인터페이스(22) 및 하나 이상의 네트워크 통신 인터페이스(26)를 포함할 수 있다. 입출력 인터페이스(22) 및 네트워크 통신 인터페이스(26)는 통신 버스(18)에 연결된다. 입출력 장치(24)는 입출력 인터페이스(22)를 통해 컴퓨팅 장치(12)의 다른 컴포넌트들에 연결될 수 있다. 예시적인 입출력 장치(24)는 포인팅 장치(마우스 또는 트랙패드 등), 키보드, 터치 입력 장치(터치패드 또는 터치스크린 등), 음성 또는 소리 입력 장치, 다양한 종류의 센서 장치 및/또는 촬영 장치와 같은 입력 장치, 및/또는 디스플레이 장치, 프린터, 스피커 및/또는 네트워크 카드와 같은 출력 장치를 포함할 수 있다. 예시적인 입출력 장치(24)는 컴퓨팅 장치(12)를 구성하는 일 컴포넌트로서 컴퓨팅 장치(12)의 내부에 포함될 수도 있고, 컴퓨팅 장치(12)와는 구별되는 별개의 장치로 컴퓨팅 장치(12)와 연결될 수도 있다.
- [0103] 이상에서 본 발명의 대표적인 실시예들을 상세하게 설명하였으나, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자는 상술한 실시예에 대하여 본 발명의 범주에서 벗어나지 않는 한도 내에서 다양한 변형이 가능함을 이해할 것이다. 그러므로 본 발명의 권리범위는 설명된 실시예에 국한되어 정해져서는 안 되며, 후술하는 특허 청구범위뿐만 아니라 이 특허청구범위와 균등한 것들에 의해 정해져야 한다.

부호의 설명

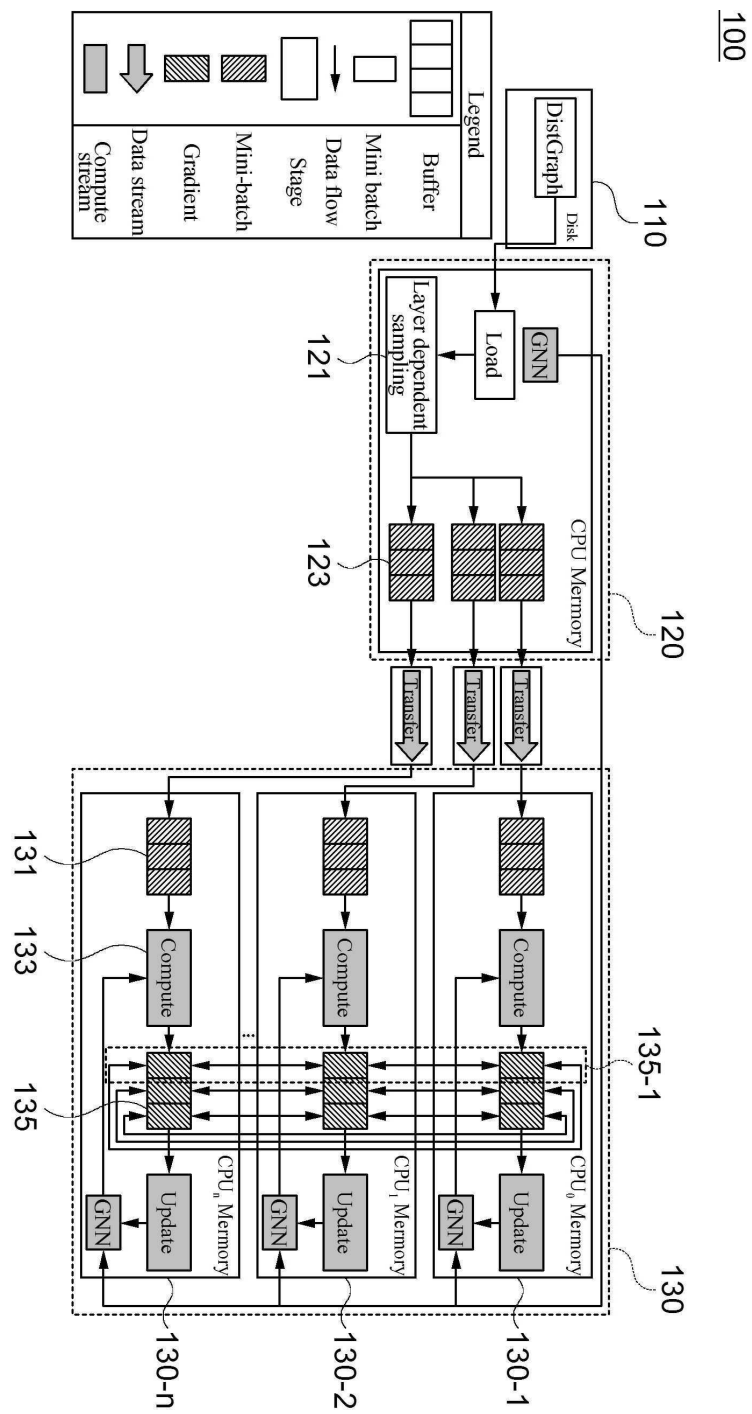
- [0105] 100: 그래프 신경망 학습 지원 장치
- 110: 인터페이스부
- 120: 미니배치 생성부
- 121: 중앙처리장치
- 123: 제 1 미니배치 버퍼
- 130: 신경망 학습부
- 131: 제 2 미니배치 버퍼
- 133: 그래픽처리장치
- 135: 그래디언트 버퍼

도면

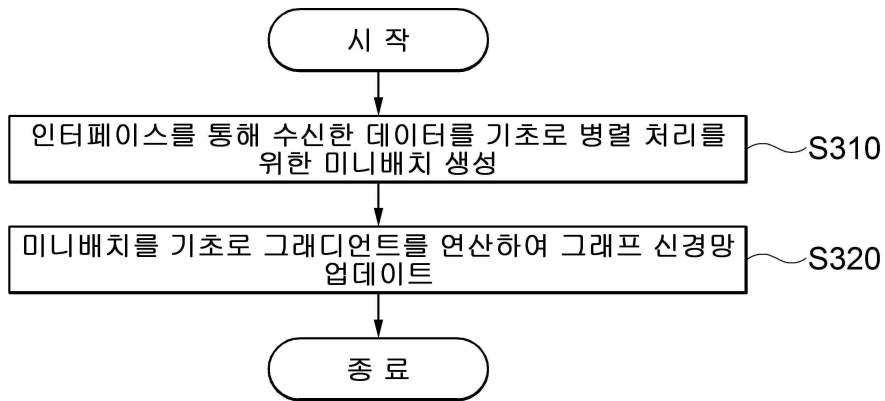
도면1



도면2



도면3



도면4

10

