(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0089279
(43) 공개일자 2025년06월18일

(51) 국제특허분류(Int. Cl.)
H04L 9/00 (2022.01)
(52) CPC특허분류
H04L 9/008 (2013.01)
(21) 출원번호 10-2023-0179015
(22) 출원일자 2023년12월11일
심사청구일자 2023년12월11일

(71) 출원인
삼성전자주식회사
경기도 수원시 영통구 삼성로 129 (매탄동)
울산과학기술원
울산광역시 울주군 언양읍 유니스트길 50
(72) 발명자
박재우
울산광역시 울주군 언양읍 유니스트길 50
이종은
울산광역시 울주군 언양읍 유니스트길 50
(뒷면에 계속)
(74) 대리인
특허법인 무한

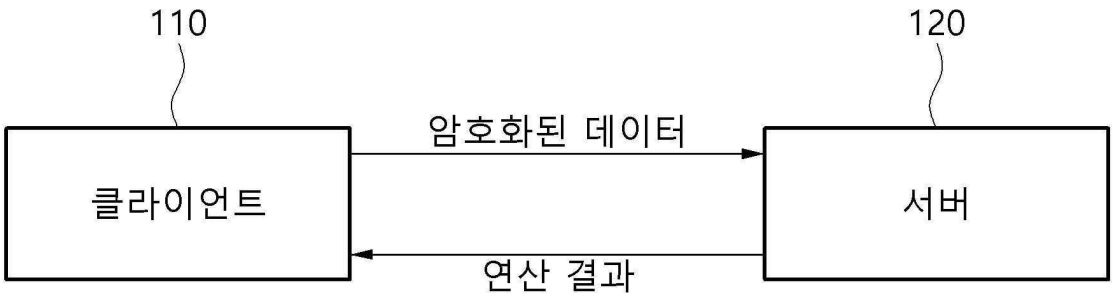
전체 청구항 수 : 총 19 항

(54) 발명의 명칭 **동형 암호 연산 방법 및 장치**

(57) 요 약

일 실시예에 따른 동형암호 연산 방법은 클라이언트로부터 동형암호 기법에 기초하여 암호화된 입력 초차원 벡터를 수신하는 단계, 상기 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 포함하는 초차원 컴퓨팅을 수행하는 단계 및 상기 초차원 컴퓨팅을 수행한 결과를 상기 클라이언트로 전달하는 단계를 포함할 수 있다.

대 표 도 - 도1



(72) 발명자
문현곤
울산광역시 울주군 언양읍 유니스트길 50
권성호
울산광역시 울주군 언양읍 유니스트길 50

명 세 서

청구범위

청구항 1

클라이언트로부터 동형암호 기법에 기초하여 암호화된 입력 초차원 벡터를 수신하는 단계;

상기 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 포함하는 초차원 컴퓨팅을 수행하는 단계; 및

상기 초차원 컴퓨팅을 수행한 결과를 상기 클라이언트로 전달하는 단계

를 포함하는 동형암호 연산 방법.

청구항 2

제1항에 있어서,

상기 초차원 컴퓨팅을 수행하는 단계는

정규화된 클래스 초차원 벡터들을 획득하는 단계; 및

상기 정규화된 클래스 초차원 벡터에 기초하여, 상기 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 수행하는 단계

를 포함하는, 동형암호 연산 방법.

청구항 3

제2항에 있어서,

상기 유사도 탐색 연산을 수행하는 단계는

상기 정규화된 클래스 초차원 벡터들을 결합하여 하나의 클래스 행렬을 생성하는 단계; 및

상기 암호화된 입력 초차원 벡터와 상기 클래스 행렬 사이의 곱셈 연산을 수행하여 상기 유사도 탐색 연산을 수행하는 단계

를 포함하는 동형암호 연산 방법.

청구항 4

제1항에 있어서,

상기 유사도 탐색 연산을 수행하는 단계는

상기 암호화된 입력 초차원 벡터를 L2norm으로 나누는 연산을 수행하지 않고, 상기 유사도 탐색 연산을 수행하는 단계

를 포함하는, 동형암호 연산 방법.

청구항 5

제1항에 있어서,

상기 유사도 탐색 연산을 수행하는 단계는

리스케일(rescale) 연산을 수행하지 않고, 상기 유사도 탐색 연산을 수행하는 단계

를 포함하는, 동형암호 연산 방법.

청구항 6

제1항에 있어서,

상기 암호화된 입력 초차원 벡터를 양자화(quantization)하는 단계

를 더 포함하는, 동형암호 연산 방법.

청구항 7

제2항에 있어서,

상기 정규화된 클래스 초차원 벡터를 양자화하는 단계;

를 더 포함하는, 동형암호 연산 방법.

청구항 8

제1항에 있어서,

상기 암호화된 입력 초차원 벡터를 수신하는 단계는

CKKS에 기초하여 암호화된 입력 초차원 벡터를 수신하는 단계

를 포함하는, 동형암호 연산 방법.

청구항 9

제1항에 있어서,

상기 암호화된 입력 초차원 벡터를 수신하는 단계는

BGV 및 BFV 중 적어도 하나에 기초하여 암호화된 입력 초차원 벡터를 수신하는 단계

를 포함하는, 동형암호 연산 방법.

청구항 10

입력 데이터를 초차원 벡터로 변환하는 단계;

동형암호 기법에 기초하여, 상기 초차원 벡터를 암호화하는 단계;

상기 암호화된 초차원 벡터를 서비스 제공 주체(service provider)로 전달하는 단계;

상기 서비스 제공 주체로부터 상기 암호화된 초차원 벡터의 유사도 탐색 연산 결과를 수신하는 단계;

상기 유사도 탐색 연산 결과를 복호화하는 단계

를 포함하는 암호화 방법.

청구항 11

하드웨어와 결합되어 제1항 내지 제10항 중 어느 하나의 항의 방법을 실행시키기 위하여 매체에 저장된 컴퓨터

프로그램.

청구항 12

클라이언트로부터 동형암호 기법에 기초하여 암호화된 입력 초차원 벡터를 수신하고,

상기 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 포함하는 초차원 컴퓨팅을 수행하고,

상기 초차원 컴퓨팅을 수행한 결과를 상기 클라이언트로 전달하는 프로세서

를 포함하는 전자 장치.

청구항 13

제12항에 있어서,

상기 프로세서는

정규화된 클래스 초차원 벡터들을 획득하고,

상기 정규화된 클래스 초차원 벡터에 기초하여, 상기 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 수행하는, 전자 장치.

청구항 14

제13항에 있어서,

상기 프로세서는

상기 정규화된 클래스 초차원 벡터들을 결합하여 하나의 클래스 행렬을 생성하고,

상기 암호화된 입력 초차원 벡터와 상기 클래스 행렬 사이의 곱셈 연산을 수행하여 상기 유사도 탐색 연산을 수행하는, 전자 장치.

청구항 15

제12항에 있어서,

상기 프로세서는

상기 암호화된 입력 초차원 벡터를 L2norm으로 나누는 연산을 수행하지 않고, 상기 유사도 탐색 연산을 수행하는, 전자 장치.

청구항 16

제12항에 있어서,

상기 프로세서는

리스케일(rescale) 연산을 수행하지 않고, 상기 유사도 탐색 연산을 수행하는, 전자 장치.

청구항 17

제13항에 있어서,

상기 프로세서는

상기 암호화된 입력 초차원 벡터를 양자화하고, 상기 정규화된 클래스 초차원 벡터를 양자화하는, 전자 장치.

청구항 18

제12항에 있어서,

상기 프로세서는

CKKS에 기초하여 암호화된 입력 초차원 벡터를 수신하는, 전자 장치.

청구항 19

제12항에 있어서,

상기 프로세서는

BGV 및 BFV 중 적어도 하나에 기초하여 암호화된 입력 초차원 벡터를 수신하는, 전자 장치.

발명의 설명

기 술 분 야

[0001] 실시예들은 동형 암호 연산 방법 및 장치에 관한 것으로, 보다 구체적으로 동형암호 기반의 기밀유지 인공지능(Privacy-Preserving Machine Learning) 시스템에 관한 것이다.

배 경 기 술

[0002] 동형 암호(homomorphic encryption)는 암호화된 데이터 사이에서 임의의 연산을 가능하게 하는 유망한 암호화 방법이다. 동형 암호를 활용하면 암호화된 데이터를 복호화하지 않고, 암호화된 상태에서 임의의 연산을 수행할 수 있을 뿐만 아니라, 격자를 기반(lattice-based)으로 하여 양자 알고리즘에 내성(resistant)이 있어 안전하다.

[0003] 동형 암호의 이러한 특성에 기반하여, 동형 암호는 다양한 인공 지능 시스템에 사용될 수 있다. 합성곱 신경망 추론을 위해 데이터를 암호화 하는 과정에서 많은 동형암호 암호문 (ciphertext) 가 생기며, 이는 많은 메모리와 네트워크 데이터 전송량을 요구한다. 이를 최소화 하기 위해 이미지와 같은 높은 차원의 데이터를 최대한 적은 양의 암호문을 사용하여 패킹(packing) 하는 방법들이 연구가 되었다.

발명의 내용

과제의 해결 수단

[0004] 일 실시예에 따른 동형암호 연산 방법은 클라이언트로부터 동형암호 기법에 기초하여 암호화된 입력 초차원 벡터를 수신하는 단계; 상기 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 포함하는 초차원 컴퓨팅을 수행하는 단계; 및 상기 초차원 컴퓨팅을 수행한 결과를 상기 클라이언트로 전달하는 단계를 포함할 수 있다.

[0005] 상기 초차원 컴퓨팅을 수행하는 단계는 정규화된 클래스 초차원 벡터들을 획득하는 단계; 및 상기 정규화된 클래스 초차원 벡터에 기초하여, 상기 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 수행하는 단계를 포함할 수 있다.

[0006] 상기 유사도 탐색 연산을 수행하는 단계는 상기 정규화된 클래스 초차원 벡터들을 결합하여 하나의 클래스 행렬을 생성하는 단계; 및 상기 암호화된 입력 초차원 벡터와 상기 클래스 행렬 사이의 곱셈 연산을 수행하여 상기 유사도 탐색 연산을 수행하는 단계를 포함할 수 있다.

[0007] 상기 유사도 탐색 연산을 수행하는 단계는 상기 암호화된 입력 초차원 벡터를 L2norm으로 나누는 연산을 수행하

지 않고, 상기 유사도 탐색 연산을 수행하는 단계를 포함할 수 있다.

[0008] 상기 유사도 탐색 연산을 수행하는 단계는 리스케일(rescale) 연산을 수행하지 않고, 상기 유사도 탐색 연산을 수행하는 단계를 포함할 수 있다.

[0009] 일 실시예에 따른 동형암호 연산 방법은 상기 암호화된 입력 초차원 벡터를 양자화(quantization)하는 단계를 더 포함할 수 있다.

[0010] 일 실시예에 따른 동형암호 연산 방법은 상기 정규화된 클래스 초차원 벡터를 양자화하는 단계를 더 포함할 수 있다.

[0011] 상기 암호화된 입력 초차원 벡터를 수신하는 단계는 CKKS에 기초하여 암호화된 입력 초차원 벡터를 수신하는 단계를 포함할 수 있다.

[0012] 상기 암호화된 입력 초차원 벡터를 수신하는 단계는 BGV 및 BFV 중 적어도 하나에 기초하여 암호화된 입력 초차원 벡터를 수신하는 단계를 포함할 수 있다.

[0013] 일 실시예에 따른 암호화 방법은 입력 데이터를 초차원 벡터로 변환하는 단계; 동형암호 기법에 기초하여, 상기 초차원 벡터를 암호화하는 단계; 상기 암호화된 초차원 벡터를 서비스 제공 주체(service provider)로 전달하는 단계; 상기 서비스 제공 주체로부터 상기 암호화된 초차원 벡터의 유사도 탐색 연산 결과를 수신하는 단계; 및 상기 유사도 탐색 연산 결과를 복호화하는 단계를 포함할 수 있다.

[0014] 일 실시예에 따른 전자 장치는 클라이언트로부터 동형암호 기법에 기초하여 암호화된 입력 초차원 벡터를 수신하고, 상기 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 포함하는 초차원 컴퓨팅을 수행하고, 상기 초차원 컴퓨팅을 수행한 결과를 상기 클라이언트로 전달하는 프로세서를 포함할 수 있다.

[0015] 상기 프로세서는 정규화된 클래스 초차원 벡터들을 획득하고, 상기 정규화된 클래스 초차원 벡터에 기초하여, 상기 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 수행할 수 있다.

[0016] 상기 프로세서는 상기 정규화된 클래스 초차원 벡터들을 결합하여 하나의 클래스 행렬을 생성하고, 상기 암호화된 입력 초차원 벡터와 상기 클래스 행렬 사이의 곱셈 연산을 수행하여 상기 유사도 탐색 연산을 수행할 수 있다.

[0017] 상기 프로세서는 상기 암호화된 입력 초차원 벡터를 L2norm으로 나누는 연산을 수행하지 않고, 상기 유사도 탐색 연산을 수행할 수 있다.

[0018] 상기 프로세서는 리스케일(rescale) 연산을 수행하지 않고, 상기 유사도 탐색 연산을 수행할 수 있다.

[0019] 상기 프로세서는 상기 암호화된 입력 초차원 벡터를 양자화하고, 상기 정규화된 클래스 초차원 벡터를 양자화할 수 있다.

[0020] 상기 프로세서는 CKKS에 기초하여 암호화된 입력 초차원 벡터를 수신할 수 있다.

[0021] 상기 프로세서는 BGV 및 BFV 중 적어도 하나에 기초하여 암호화된 입력 초차원 벡터를 수신할 수 있다.

도면의 간단한 설명

[0022] 도 1은 일 실시예에 따른 기밀 유지 인공지능 시스템을 설명하기 위한 도면이다.

도 2는 일 실시예에 따른 클라이언트의 동작 방법을 설명하기 위한 도면이다.

도 3은 일 실시예에 따른 서버의 동작 방법을 설명하기 위한 도면이다.

도 4는 일 실시예에 따른 양자화 방법을 설명하기 위한 도면이다.

도 5는 일 실시예에 따른 동형 암호 연산 방법에 따른 추론 속도를 설명하기 위한 도면이다.

도 6은 일 실시예에 따른 서버의 동작 방법을 설명하기 위한 도면이다.

도 7은 일 실시예에 따른 클라이언트의 동작 방법을 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0023] 본 명세서에서 개시되어 있는 특정한 구조적 또는 기능적 설명들은 단지 기술적 개념에 따른 실시예들을 설명하

기 위한 목적으로 예시된 것으로서, 실제로 구현된 형태는 다양한 다른 모습을 가질 수 있으며 본 명세서에 설명된 실시예로만 한정되지 않는다.

[0024] 제1 또는 제2 등의 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 이런 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 이해되어야 한다. 예를 들어 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소는 제1 구성요소로도 명명될 수 있다.

[0025] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성요소들 간의 관계를 설명하는 표현들, 예를 들어 "~간의"와 "바로~간의" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0026] 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 실시된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0027] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 해당 기술분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0028] 실시예들은 퍼스널 컴퓨터, 랩톱 컴퓨터, 태블릿 컴퓨터, 스마트 폰, 텔레비전, 스마트 가전 기기, 지능형 자동차, 키오스크, 웨어러블 장치 등 다양한 형태의 제품으로 구현될 수 있다. 이하, 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0029] 도 1은 일 실시예에 따른 기밀 유지 인공지능 시스템을 설명하기 위한 도면이다.

[0030] 도 1을 참조하면, 일 실시예에 따른 기밀 유지 인공지능 시스템은 클라이언트(110) 및 서버(120)를 주체로 포함할 수 있다.

[0031] 기밀 유지 인공지능 시스템은 클라이언트(110)가 보유한 데이터를 서버(120)에게 직접적으로 노출시키지 않으면서 서버(120)가 클라이언트(110)로 인공지능 서비스를 제공하는 시스템일 수 있다.

[0032] 클라이언트(110)는 서버(120)로부터 인공지능 서비스를 제공 받는 주체로, 서비스 이용 주체, 서비스 이용자, 데이터 소유자 등으로 지칭될 수 있다. 클라이언트(110)는 클라이언트 단말을 통해 자신의 데이터(예를 들어, 이미지)를 동형 암호 기법에 기초하여 암호화하고, 암호화된 데이터를 서버(120)로 전달할 수 있다. 클라이언트 단말은 사용자 단말로 지칭될 수 있다.

[0033] 동형 암호(Homomorphic encryption, HE)는 암호화된 데이터를 복호화 없이 연산할 수 있는 암호기술로, 동형 암호화된 상태에서 각종 연산을 했을 때 그 결과가 암호화하지 않은 상태의 연산 결과와 동일하다. 동형 암호는 암호화된 상태로 데이터를 가공할 수 있기 때문에 데이터 산업에서 발생하는 프라이버시 문제를 해결할 수 있다.

[0034] 서버(120)는 클라이언트(110)로부터 암호화된 데이터를 수신하여, 암호화된 데이터에 대응하는 인공 지능 연산 결과를 클라이언트(110)에게 전달할 수 있으며, 서버(120)는 서비스 제공자(service provider), 서비스 제공 주체 등으로 지칭될 수 있다.

[0035] 서버(120)는 초차원 컴퓨팅에 기반 인공지능 서비스를 제공할 수 있다. 초차원 컴퓨팅 기반 인공지능은 딥 뉴럴 네트워크(DNN, Deep Neural Network) 방식의 인공지능에 비해 저전력, 저비용 알고리즘일 수 있다. 동형암호 기법에 기초하여 암호화된 데이터를 처리할 때 딥 뉴럴 네트워크에 대신 초차원 컴퓨팅구조를 사용할 경우 연산량과 실행시간을 대폭 절감할 수 있다.

[0036] 보다 구체적으로, 동형암호 연산은 덧셈과 곱셈, 그리고 같은 암호 내에서 순서를 바꾸는 회전(rotate) 연산들을 지원하며, 덧셈, 곱셈, 그리고 회전 순서대로 실행시간이 증가한다.

[0037] 동형암호는 미리 지정된 다항식 차수(degree of polynomial)를 파라미터로 가지며, 이 차수는 2의 거듭제곱 꼴이어야만 한다. 다항식 차수가 높을수록 모든 연산의 실행시간이 기하급수적으로 증가하는 단점이 있으나, 다항식 차수가 높을수록 더 다양한 연산을 지원 가능하게 되고 계산의 정밀도가 늘어난다는 장점이 있다.

[0038] 동형암호 연산은 미리 지정된 최대곱셈횟수 만큼의 동형암호 곱셈이 가능하며, 이 최대곱셈횟수가 증가할 수록 동형암호 연산 1회에 걸리는 시간이 기하급수적으로 증가한다. 동시에 이 최대곱셈횟수는 다항식 차수에 제한이 되며, 차수가 높을수록 최대곱셈횟수 또한 늘어난다.

[0039] 즉, 동형암호 곱셈 연산의 수가 적을수록 동형암호 연산이 빨라질 수 있다. 복잡한 비선형 활성화 함수와 여러 번의 곱셈과 덧셈이 필요한 딥 뉴럴 네트워크를 기반으로 하는 종래의 인공 지능 시스템은 실제 서비스에 사용되기 매우 느릴 수 있다(예를 들어, 수초~ 3000초). 반면, 서포트 벡터 머신과 같은 또 다른 종래의 인공지능 시스템은 딥 뉴럴 네트워크보다 요구되는 연산량이 적으나 분류 정확도가 매우 낮고 사용성이 제한된다는 단점이 있다.

[0040] 아래에서 상세히 설명하겠지만, 일 실시예에 따른 초차원 컴퓨팅 구조의 인공지능 시스템은 추론을 위해 유사도 검색 알고리즘을 사용하며, 이는 오직 한번의 동형암호-평서문(ciphertext-plaintext) 곱셈과 동형암호 회전, 그리고 덧셈만을 요구하기 때문에 동형암호 위에서의 연산이 매우 빠르며, 동시에 다른 인공지능 알고리즘과 비교하였을 시 딥 뉴럴 네트워크에 준하는 분류 정확도를 가질 수 있다.

[0041] 따라서, 일 실시예에 따른 초차원 컴퓨팅 구조의 인공지능 시스템은 엄청나게 많은 컴퓨팅 자원을 요구하는 기존의 딥 뉴럴 네트워크 기반 기밀보장 인공지능 시스템 대신 일상적으로 사용되는 하드웨어에서 실시간으로 서비스가 가능한 기밀보장 인공지능 서비스를 제공할 수 있다. 예를 들어, 일 실시예에 따른 초차원 컴퓨팅 구조의 인공지능 시스템은 얼굴인식 혹은 마스크 탐지 등 사용자 데이터의 기밀 유지가 중요한 상황에서 유용하게 사용될 수 있다.

[0042] 도 2는 일 실시예에 따른 클라이언트의 동작 방법을 설명하기 위한 도면이다. 도 1을 참조하여 설명한 내용은 도 2에 동일하게 적용될 수 있다.

[0043] 도 2를 참조하면, 일 실시예에 따른 클라이언트(110)는 입력 데이터(210)를 초차원 벡터로 변환할 수 있다. 입력 데이터는 예를 들어, 사용자의 개인정보 데이터일 수 있다. 클라이언트(110)는 초차원 인코더(hyperdimensional encoder)를 사용하여 입력 데이터를 초차원 벡터(HV)로 변환할 수 있고, 변환된 초차원 벡터는 쿼리 초차원 벡터(query HV)(220)로 지칭될 수 있다.

[0044] 보다 구체적으로, 입력 데이터는 인코딩 함수(예를 들어, $\phi : X \rightarrow H$)를 기반으로 입력 공간에서 초차원 벡터 공간으로 매핑될 수 있다. 선형 분류기는 복잡한 데이터들 사이에서는 제대로 동작하지 못할 수 있다. 그러나, 입력 데이터를 더 높은 차원으로 매핑하면 입력 데이터를 나누는 초평면을 성공적으로 찾을 수 있다. 초차원 컴퓨팅은 입력 데이터를 초차원 벡터로 잘 표현하고, 분류 프로세서를 단순화하는 것을 목표로 한다. 인코더는 입력 데이터를 고차원(예를 들어, 4k ~ 8k)의 쿼리 초차원 벡터로 인코딩할 수 있다.

[0045] 클라이언트(110)는 동형암호 기법을 사용하여 초차원 벡터를 암호화(encryption)할 수 있다. 동형 암호에서 암호문(ciphertext)들을 이용한 연산의 결과는 새로운 암호문이 되고, 암호문을 복호화함으로써 획득된 평문(plaintext)은 암호화하기 전의 원래 데이터의 연산 결과와 동일할 수 있다. 이하에서, 암호화된 데이터(encrypted data) 또는 암호문은 사이퍼텍스트(ciphertext)로 지칭될 수 있다. 사이퍼텍스트는 다항식 또는 다항식을 포함하는 벡터의 형태를 가질 수 있다.

[0046] 클라이언트 단말은 기밀 유지 인공지능 시스템에서 입력 데이터를 암호화하는 암호화 과정을 수행할 수 있다. 클라이언트 단말은 칩(chip) 형태로 구현되어 동형 암호를 활용하는 하드웨어 가속기에 탑재될 수 있다. 클라이언트 단말은 칩 형태 또는 소프트웨어 형태로 구현되어 다양한 연산 장치의 메모리 사용을 감소시킬 수 있다. 클라이언트 단말은 동형 암호 연산의 연산량을 감소시킴으로써, 서버의 전체 연산량을 감소시킬 수 있다.

[0047] 클라이언트 단말은 RLWE 문제 기반 모든 동형 암호에 적용될 수 있다. 클라이언트 단말은 동형 암호를 적용하는 모든 장치와 서비스에서 입력값을 암호화하는 암호화 과정에 구현될 수 있다.

[0048] 클라이언트 단말은 PC(personal computer), 데이터 서버, 또는 휴대용 장치 내에 구현될 수 있다. 휴대용 장치는 랩탑(laptop) 컴퓨터, 이동 전화기, 스마트 폰(smart phone), 태블릿(tablet) PC, 모바일 인터넷 디바이스(mobile internet device(MID)), PDA(personal digital assistant), EDA(enterprise digital assistant), 디지털 스틸 카메라(digital still camera), 디지털 비디오 카메라(digital video camera), PMP(portable

multimedia player), PND(personal navigation device 또는 portable navigation device), 휴대용 게임 콘솔(handheld game console), e-북(e-book), 또는 스마트 디바이스(smart device)로 구현될 수 있다. 스마트 디바이스는 스마트 와치(smart watch), 스마트 밴드(smart band), 또는 스마트 링(smart ring)으로 구현될 수 있다.

[0049] 클라이언트(110)는 암호화된 쿼리 초차원 벡터(230)를 서버(120)에게 전달할 수 있다. 서버(120)는 암호화된 쿼리 초차원 벡터(230)로 연산을 한 후 연산 결과를 클라이언트에게 돌려줄 수 있다. 서버(120)는 암호화된 쿼리 초차원 벡터(230)로 초차원 컴퓨팅 연산을 수행할 수 있다. 예를 들어, 서버(120)는 암호화된 쿼리 초차원 벡터(230)로 유사도 탐색 연산을 수행할 수 있다.

[0050] 클라이언트(110)는 서버(120)로부터 수신한 연산 결과를 복호화할 수 있다. 클라이언트(110)는 복호화된 데이터(240)에 기초하여 서버(120)가 제공한 인공 지능 결과(250)를 확인할 수 있다. 예를 들어, 클라이언트(110)는 유사도 탐색 결과를 복호화하여 인공지능의 분류 결과(예를 들어, "7")를 확인할 수 있다.

[0051] 클라이언트 단말은 PIM (processing in memory)을 포함할 수 있다. 클라이언트 단말은 PIM을 이용하여 입력 데이터를 초차원 벡터로 변환하는 연산(이하, 초차원 벡터 인코딩 연산), 초차원 벡터를 암호화하는 연산(이하, 동형암호 암호화 연산)을 수행할 수 있다.

[0052] 초차원 벡터 인코딩 연산은 입력 데이터를 초차원 벡터로 변환하는 연산으로, 예를 들어, 입력 데이터가 사진인 경우에는 사진에 무작위 행렬(random matrix)을 곱하여 높은 차원을 가진 벡터로 변환할 수 있다. PIM은 행렬곱 연산을 지원하기 때문에, 클라이언트 단말은 PIM에서 초차원 벡터 인코딩 연산을 수행할 수 있다.

[0053] 동형암호 암호화 연산은 행렬곱 연산과FFT(Fast Fourier Transform) 연산 또는 NTT(Number Theoretic Transform) 연산을 포함할 수 있다. NTT는 고속 푸리에 변환의 복소수 연산을 정수 모듈러 연산으로 변형한 알고리즘일 수 있다. PIM에서 행렬곱 연산, FFT 연산, NTT 연산 모두 수행할 수 있다.

[0054] 도 3은 일 실시예에 따른 서버의 동작 방법을 설명하기 위한 도면이다. 도 1 내지 도 2를 참조하여 설명한 내용은 도 3에 동일하게 적용될 수 있다.

[0055] 도 3을 참조하면, 일 실시예에 따른 서버(120)는 클라이언트(110)로부터 암호화된 쿼리 초차원 벡터(230)를 수신할 수 있다. 서버(120)는 암호화된 쿼리 초차원 벡터(230)로 초차원 컴퓨팅 연산을 수행할 수 있다. 예를 들어, 서버(120)는 초차원 컴퓨팅을 통해 분류(classification) 태스크를 처리할 수 있다. 서버(120)는 레이블링된 클래스 초차원 벡터(310)와 클라이언트(110)로부터 수신한 암호화된 쿼리 초차원 벡터(230) 사이의 유사도 탐색(similarity search) 연산을 수행할 수 있다. 예를 들어, 서버(120)는 레이블링된 클래스 초차원 벡터(310)와 암호화된 쿼리 초차원 벡터(230) 사이의 코사인 유사도에 기초한 유사도 탐색 연산을 수행할 수 있다. 서버(120)는 유사도 탐색 연산 결과에 기초하여, 유사도 점수(320)를 결정할 수 있다.

[0056] 일 실시예에 따른 유사도 탐색 연산은 딥 뉴럴 네트워크의 연산에 비해 훨씬 빠를 수 있다. 나아가, 일 실시예에 따른 서버(120)는 동형암호 기반으로 사용자 데이터를 보호하는 초차원 컴퓨팅 시스템에서 나눗셈 연산과 리스케일(rescale) 연산까지 생략된 유사도 탐색 연산을 수행할 수 있다.

[0057] 기존 연구들은 유사도 탐색을 주로 저전력 하드웨어 뉴로모픽 하드웨어 혹은 PIM (Processing-In-Memory) 하드웨어를 목표로 하며, 각 목표 하드웨어에 맞게 최적화를 시켜 사용하나 동형암호에 최적화를 시킨 유사도 탐색 알고리즘은 존재하지 않는다. 반면에, 일 실시예에 따른 유사도 판단 연산 방법은 동형암호 연산의 특성을 매우 정밀하게 분석해 곱셈과 비선형 연산이 요구되는 유사도 연산을 동형암호에 알맞게 최적화 시키며, 이는 연산 결과의 정확도를 유지한 채 30%의 연산 속도 향상을 가질 수 있다.

[0058] 아래 수학식 1은 초차원 컴퓨팅 구조 기반의 인공지능 분류기에 사용되는 유사도 검색 알고리즘을 나타낸다.

수학식 1

[0059]
$$\delta(\vec{\mathcal{H}}_q, \vec{\mathcal{C}}^i) = \frac{\vec{\mathcal{H}}_q^{\intercal} \cdot \vec{\mathcal{C}}^i}{\|\vec{\mathcal{H}}_q\| \cdot \|\vec{\mathcal{C}}^i\|}$$

[0060] 수학식 1을 참조하면, 종래의 유사도 탐색 알고리즘은 길이가 매우 긴 초차원의 벡터(예를 들어, 2000 차원 이상) 내적 결과(inner product)를 두 벡터의 L2 norm 으로 나눈 결과를 반환한다. 수학식 1에서, $\vec{\mathcal{H}}_q$ 는 분류 결과를 알고자 하는 입력 초차원 벡터(query HV)이며, $\vec{\mathcal{C}^l}$ 는 인공지능 모델의 가중치로, 클래스 초차원 벡터(class HV)이다.

[0061] 대부분의 동형암호 기법은 나눗셈을 지원하지 않을 뿐더러, 곱셈 연산 마저도 느리기 때문에, 일 실시예에 따른 유사도 탐색 연산 방법은 곱셈을 피하기 위해 클래스 초차원 벡터를 사전에 정규화 해서 저장할 수 있다. 또한, 분류 결과를 알기 위해서는 여러 클래스 초차원 벡터와의 유사도 결과 중 최대값 만을 알면 되기 때문에 유사도 결과 사이의 절대적인 값은 중요치 않으며, 상대적인 대소만 보존된다면 추론 결과를 올바르게 계산하는 데 문제가 없을 수 있다. 이를 이용하여, 일 실시예에 따른 유사도 탐색 연산 방법은 입력 초차원 벡터를 L2 norm 으로 나누는 연산 또한 생략할 수 있다. 이를 수학식으로 표현하면, 아래 수학식 2와 같다.

수학식 2

[0062] $$\delta(\vec{\mathcal{H}}_q, \vec{\mathcal{C}^l}) = \frac{\vec{\mathcal{H}}_q^{\intercal} \cdot \vec{\mathcal{C}^l}}{\|\vec{\mathcal{H}}_q\| \cdot \|\vec{\mathcal{C}^l}\|} = \frac{\vec{\mathcal{H}}_q^{\intercal} \cdot \vec{\mathcal{C}^l}}{\|\vec{\mathcal{H}}_q\|}$$

[0063] 같은 이유로 CKKS 동형암호의 경우, 일 실시예에 따른 유사도 탐색 연산 방법에서는 곱셈 후 곱셈의 소수점 위치를 보존하기 위한 리스케일(rescale) 연산 또한 생략할 수 있다.

[0064] 효율적인 초차원 컴퓨팅의 유사도 탐색을 위해 클래스 초차원 벡터를 결합해 수학식 3과 같이 하나의 클래스 행렬을 구성할 수 있고, 일 실시예에 따른 유사도 탐색 연산은 수학식 4와 같이 하나의 행렬 곱으로 단순화될 수 있다.

수학식 3

[0065] $$M = \begin{bmatrix} \vec{\mathcal{C}^{l1}} & \vec{\mathcal{C}^{l2}} & \cdots & \vec{\mathcal{C}^{lL}} \end{bmatrix}$$

수학식 4

[0066] $$\vec{\delta}(\vec{\mathcal{H}}_q, M) = \vec{\mathcal{H}}_q^{\intercal} \cdot M$$

[0067] 아래 표 1을 참조하면, 일 실시예에 따른 유사도 탐색 연산 방법에 따르면, 연산 정확도를 유지하는 동시에 기존의 유사도 탐색 알고리즘 대비 추론 속도가 향상(예를 들어, 30%)됨을 알 수 있다.

표 1

[0068]

Method	Inference Latency (ms)
Without Normalization and with Rescale†	257.37 (×1.30)
Without Normalization (Rescale Skipped)	214.75 (×1.09)
With Normalization (Rescale Skipped)	197.41 (×1)

Note. 1. $N = 8192$ and $D = 4096$.
2. †Requires a longer modulus chain.

[0069] 서버(120)는 초차원 컴퓨팅 연산의 수행 결과를 클라이언트(110)로 전달할 수 있다. 예를 들어, 서버(120)는 유사도 점수(320)를 클라이언트(110)로 전달할 수 있다.

[0070] 서버(120)는 클라이언트(110)와 네트워크를 통해 통신하여 명령, 코드, 파일, 컨텐츠, 서비스 등을 제공하는 컴퓨터 장치 또는 복수의 컴퓨터 장치들로 구현될 수 있다. 서버(120)는 클라이언트(110)로 인공지능 서비스를 제공할 수 있다. 예를 들어, 서버(120)는 분류 태스크 서비스 등을 제공할 수 있다. 서버(120)가 제공하는 서비스는 위 예시에 제한되지 않고, 인공지능에 연계된 다양한 서비스를 포함할 수 있다.

[0071] 서버(120)와 클라이언트(110)의 통신 방식은 제한되지 않으며, 네트워크가 포함할 수 있는 통신망(일례로, 이동통신망, 유선 인터넷, 무선 인터넷, 방송망)을 활용하는 통신 방식뿐만 아니라 기기들간의 근거리 무선 통신 역시 포함될 수 있다. 예를 들어, 네트워크는, PAN(personal area network), LAN(local area network), CAN(campus area network), MAN(metropolitan area network), WAN(wide area network), BBN(broadband network), 인터넷 등의 네트워크 중 하나 이상의 임의의 네트워크를 포함할 수 있다. 또한, 네트워크는 버스 네트워크, 스타 네트워크, 링 네트워크, 메쉬 네트워크, 스타-버스 네트워크, 트리 또는 계층적(hierarchical) 네트워크 등을 포함하는 네트워크 토폴로지 중 임의의 하나 이상을 포함할 수 있으나, 이에 제한되지 않는다.

[0072] 서버(120)는 PIM (processing in memory)을 포함할 수 있다. 서버(120)는 PIM을 이용하여 전술한 동형암호에 최적화된 초차원 컴퓨팅 연산(예를 들어, 유사도 탐색 연산)을 수행할 수 있다.

[0073] 도 4는 일 실시예에 따른 양자화 방법을 설명하기 위한 도면이다. 도 1 내지 도 3을 참조하여 설명한 내용은 도 4에 동일하게 적용될 수 있다.

[0074] 일 실시예에 따른 기밀 유지 인공지능 시스템은 클래스 초차원 벡터와 쿼리 초차원 벡터를 양자화(예를 들어, 32bit 부동 소수점 타입 대신 32bit/16bit 정수형 타입 사용)할 수 있다.

[0075] 초차원 컴퓨팅 구조에서의 양자화는 가속기(예를 들어, PIM 가속기) 하드웨어에서의 성능 향상을 위해 주로 곱셈의 연산 정밀도를 줄이는 것을 목표로 한다. 그러나 종래 기술 중에 동형암호 연산의 다항식 차수를 낮춰 궁극적으로는 동형암호 연산의 속도를 높이기 위하여 양자화를 하거나, 동형암호의 부정확한 암호화된 연산 내에서 인공지능의 분류 정확도를 높이기 위해 양자화를 적용하는 기법은 없다.

[0076] 동형암호 연산은 파라미터에 따라 지원 가능한 연산의 종류가 복잡할수록 더 느린 연산 속도와 높은 연산 정확도를 가지는 특성이 있다. 일 실시예에 따른 기밀 유지 인공지능 시스템은 빠른 연산을 위하여 낮은 연산 정밀도에서도 높은 분류 정확도를 유지하게 해 주는 양자화 방법을 제공할 수 있다. 이를 통해, 기밀 유지 인공지능 시스템은 동형암호의 보안 수준을 타협하지 않으면서 연산 속도를 가속할 수 있다.

[0077] 동형암호의 다항식 차수가 줄면 연산 속도가 빨라진다. 그러나 동시에 제한된 연산의 종류 뿐만 아니라 제한된 연산 정밀도를 가지며, 연산을 거듭할수록 오차가 쌓이는 동형암호의 특성상 복호화 시 계산 정확도가 떨어질 수 있다.

[0078] 도 4를 참조하면, 도면(400)은 연산 정밀도를 줄였을 때 초차원 컴퓨팅 기반의 인공지능 분류기 정확도의 감소를 도시한 그래프이다. 그래프(410)는 동형암호의 연산정밀도가 fp32일 때, 다항식 차수에 따른 추론 정확도를 도시한 그래프이고, 그래프(420)는 동형암호의 연산정밀도가 int16일 때, 다항식 차수에 따른 추론 정확도를 도시한 그래프이고, 그래프(430)는 동형암호의 연산정밀도가 int8일 때, 다항식 차수에 따른 추론 정확도를 도시한 그래프이다. 제1 영역(440)은 동형 암호 다항식 차수 2048로 표현이 가능한 영역을 의미하고, 제2 영역(450)은 동형 암호 다항식 차수 4096으로 표현이 가능한 영역을 의미하고, 제3 영역(460)은 동형 암호 다항식 차수 8192로 표현이 가능한 영역을 의미한다.

[0079] 낮은 연산 정밀도에서 높은 분류 정확도를 확보하기 위해서, 일 실시예에 따른 서버는 인공지능 모델에 해당하는 클래스 초차원 벡터를 양자화할 수 있다. 일 실시예에 따른 클라이언트는 암호화된 입력 초차원 벡터를 양자화할 수 있다. 예를 들어, 그래프(410)와 그래프(430)를 참조하면, 클래스 초차원 벡터를 int8로 양자화 시켰을 시 극단적으로 낮은 연산 정확도에서도 기존의 부동소수점(fp32) 대비 월등히 높은 분류 정확도를 가질 수 있다.

[0080] 도 5는 일 실시예에 따른 동형 암호 연산 방법에 따른 추론 속도를 설명하기 위한 도면이다.

[0081] 도 5를 참조하면, 그래프(500)의 x축은 추론 레이턴시(latency)(ms 단위)를, y축은 추론 정확도(%)를 의미하고,

그래프에 기재된 (a, b)에서 a는 동형암호 다항식 차수를, b는 양자화 수준(연산 정밀도)을 의미하며, 2k, 4k, 8k, 10k는 초차원 벡터의 차원 수를 의미할 수 있다. 양자화 수준(연산 정밀도)과 관련하여, q32/c16은 fp32를 int 16으로 양자화한 것을 의미하며, q32/c8은 fp32를 int 8로 양자화한 것을 의미한다.

[0082] 그래프(500)를 참조하면, 4096의 동형암호 다항식 차수와 4096의 초차원 벡터 차원 그리고 16비트 정수로 양자화된 클래스 초차원 벡터 사용 시 딥 뉴럴 네트워크와 비슷한 수준의 분류 정확도(97.10%)를 가지면서 0.083초 내에 분류를 수행할 수 있음을 알 수 있다.

[0083] 도 6은 일 실시예에 따른 서버의 동작 방법을 설명하기 위한 도면이다. 도 1 내지 도 5를 참조하여 설명한 내용은 도 6에 동일하게 적용될 수 있다.

[0084] 설명의 편의를 위해, 단계들(610 내지 630)은 도 1에 도시된 서버(120)를 사용하여 수행되는 것으로 기술된다. 그러나 이 단계들(610 내지 630)은 어떤 다른 적절한 전자 기기를 통해, 그리고 어떤 적절한 시스템 내에서도 사용될 수 있을 것이다.

[0085] 나아가, 도 6의 동작은 도시된 순서 및 방식으로 수행될 수 있지만, 도시된 실시예의 사상 및 범위를 벗어나지 않으면서 일부 동작의 순서가 변경되거나 일부 동작이 생략될 수 있다. 도 6에 도시된 다수의 동작은 병렬로 또는 동시에 수행될 수 있다.

[0086] 단계(610)에서, 일 실시예에 따른 서버(예를 들어, 도 1을 참조하여 설명한 서버(120))는 클라이언트로부터 동형암호 기법에 기초하여 암호화된 입력 초차원 벡터를 수신할 수 있다. 서버는 CKKS에 기초하여 암호화된 입력 초차원 벡터를 수신할 수 있다. 또는, 서버는 BGV 및 BFV 중 적어도 하나에 기초하여 암호화된 입력 초차원 벡터를 수신할 수 있다.

[0087] 단계(620)에서, 서버는 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 포함하는 초차원 컴퓨팅을 수행할 수 있다. 서버는 정규화된 클래스 초차원 벡터들을 획득하고, 정규화된 클래스 초차원 벡터에 기초하여, 암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 수행할 수 있다. 서버는 정규화된 클래스 초차원 벡터들을 결합하여 하나의 클래스 행렬을 생성하고, 암호화된 입력 초차원 벡터와 클래스 행렬 사이의 곱셈 연산을 수행하여 유사도 탐색 연산을 수행할 수 있다.

[0088] 서버는 유사도 탐색 연산을 수행 시에, 암호화된 입력 초차원 벡터를 L2norm으로 나누는 연산을 생략할 수 있다. 또한, 서버는 유사도 탐색 연산을 수행 시에, 리스케일(rescale) 연산을 생략할 수 있다. 서버는 암호화된 입력 초차원 벡터 및/또는 정규화된 클래스 초차원 벡터를 양자화할 수 있다.

[0089] 단계(630)에서 서버는 초차원 컴퓨팅을 수행한 결과를 클라이언트로 전달할 수 있다.

[0090] 서버는 초차원 컴퓨팅을 이용하여, 저비용으로 현실적인 시간 내에 실행이 가능한 동형암호 기반 기밀보장 인공지능 서비스를 제공할 수 있다. 일 실시예에 따른 서버는 초차원 컴퓨팅 구조를 동형암호에 최적화시켜 종래의 딥 뉴럴 네트워크 기반의 동형암호 알고리즘 대비, CPU 실행 기준, 26~3000배 실행시간을 단축시킬 수 있다.

[0091] 도 7은 일 실시예에 따른 클라이언트의 동작 방법을 설명하기 위한 도면이다. 도 1 내지 도 6을 참조하여 설명한 내용은 도 7에 동일하게 적용될 수 있다.

[0092] 설명의 편의를 위해, 단계들(710 내지 750)은 도 1에 도시된 클라이언트(110)를 사용하여 수행되는 것으로 기술된다. 그러나 이 단계들(710 내지 750)은 어떤 다른 적절한 전자 기기를 통해, 그리고 어떤 적절한 시스템 내에서도 사용될 수 있을 것이다.

[0093] 나아가, 도 7의 동작은 도시된 순서 및 방식으로 수행될 수 있지만, 도시된 실시예의 사상 및 범위를 벗어나지 않으면서 일부 동작의 순서가 변경되거나 일부 동작이 생략될 수 있다. 도 7에 도시된 다수의 동작은 병렬로 또는 동시에 수행될 수 있다.

[0094] 단계(710)에서, 일 실시예에 따른 클라이언트는 입력 데이터를 초차원 벡터로 변환할 수 있다. 클라이언트는 인코더를 이용하여 입력 데이터를 초차원 벡터로 변환할 수 있다. 인코더는 작위성에 기반을 한 인코더일 수 있으나, 이에 한정되지는 않는다. 예를 들어, 인코더는 무작위성에 기반하지 않은 인코터일 수도 있다.

[0095] 단계(720)에서, 클라이언트는 동형암호 기법에 기초하여, 초차원 벡터를 암호화할 수 있다. 클라이언트는 CKKS 기반의 동형암호에 기초하여 초차원 벡터를 암호화할 수 있다. 그러나, 암호화 방법이 CKKS 기반의 동형암호로 한정되는 것은 아니다. 예를 들어, 일 실시예에 따른 암호화 방법은 동시에 여러 곱셈과 덧셈(batching)을 지원하는 모든 동형암호(예를 들어, BGV, BFV, NTRU) 기반의 암호화 방법을 포함할 수 있다.

[0096] 단계(730)에서, 클라이언트는 암호화된 초차원 벡터를 서버(서비스 제공 주체)로 전달할 수 있다.

[0097] 단계(740)에서, 클라이언트는 서버로부터 암호화된 초차원 벡터의 유사도 탐색 연산 결과를 수신할 수 있다. 예를 들어, 클라이언트는 서버로부터 유사도 탐색 연산 결과(예를 들어, 유사도 점수)를 수신할 수 있다.

[0098] 단계(750)에서, 클라이언트는 유사도 탐색 연산 결과를 복호화할 수 있다. 클라이언트는 유사도 탐색 결과를 복호화하여 인공지능의 서비스에서 제공하는 태스크 결과(예를 들어, 분류 결과)를 확인할 수 있다.

[0099] 이상에서 설명된 실시예들은 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치, 방법 및 구성요소는, 예를 들어, 프로세서, 콘트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(OS) 및 운영 체제 상에서 수행되는 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(processing element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 콘트롤러를 포함할 수 있다. 또한, 병렬 프로세서(parallel processor)와 같은, 다른 처리 구성(processing configuration)도 가능하다.

[0100] 소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 가상 장치(virtual equipment), 컴퓨터 저장 매체 또는 장치, 또는 전송되는 신호 파(signal wave)에 영구적으로, 또는 일시적으로 구체화(embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

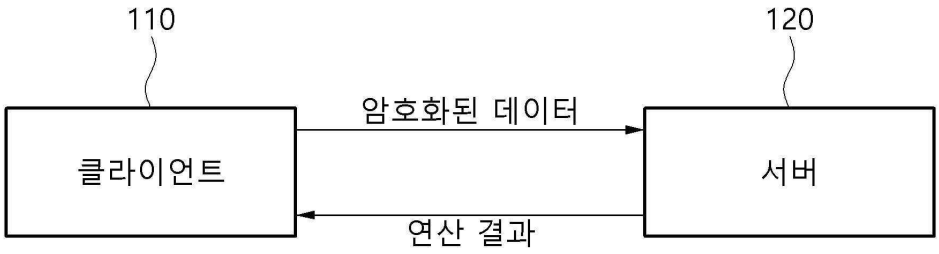
[0101] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), 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.

[0102] 이상과 같이 실시예들이 비록 한정된 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기를 기초로 다양한 기술적 수정 및 변형을 적용할 수 있다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

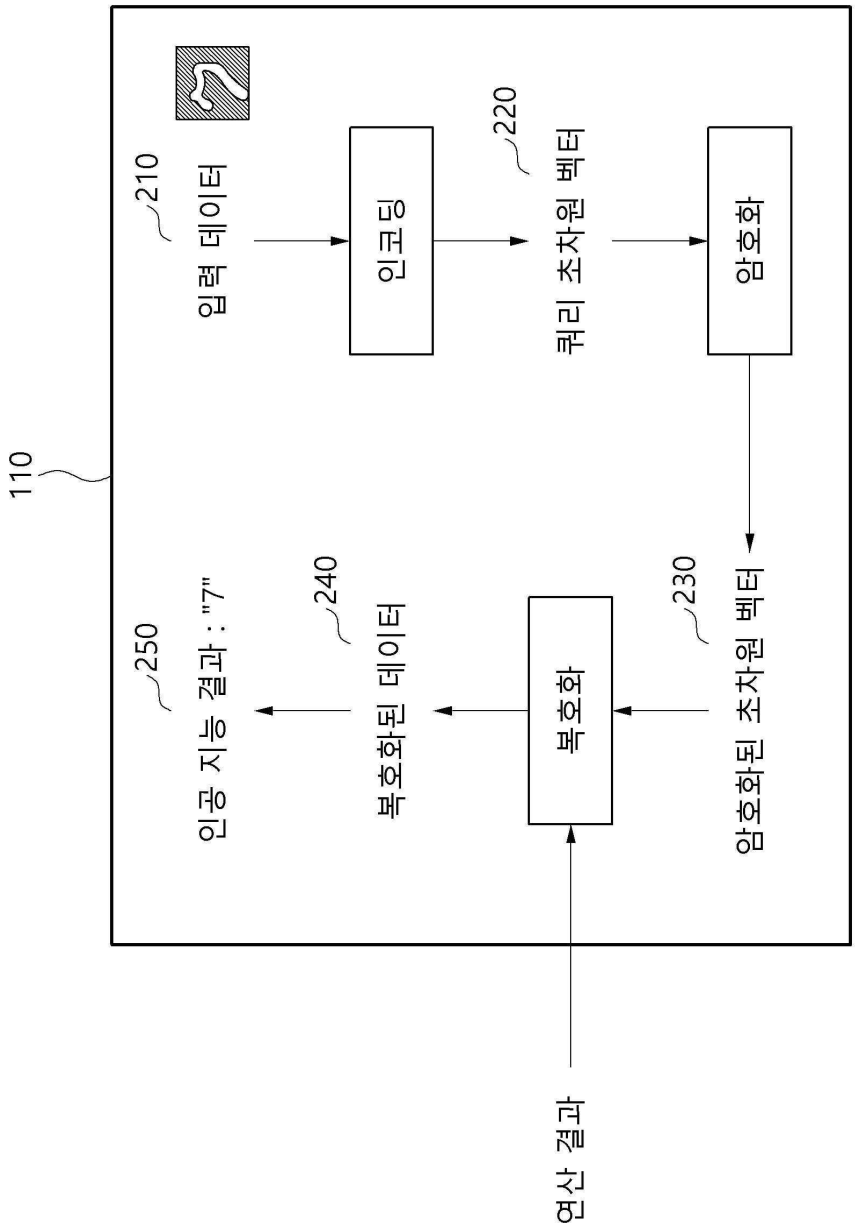
[0103] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

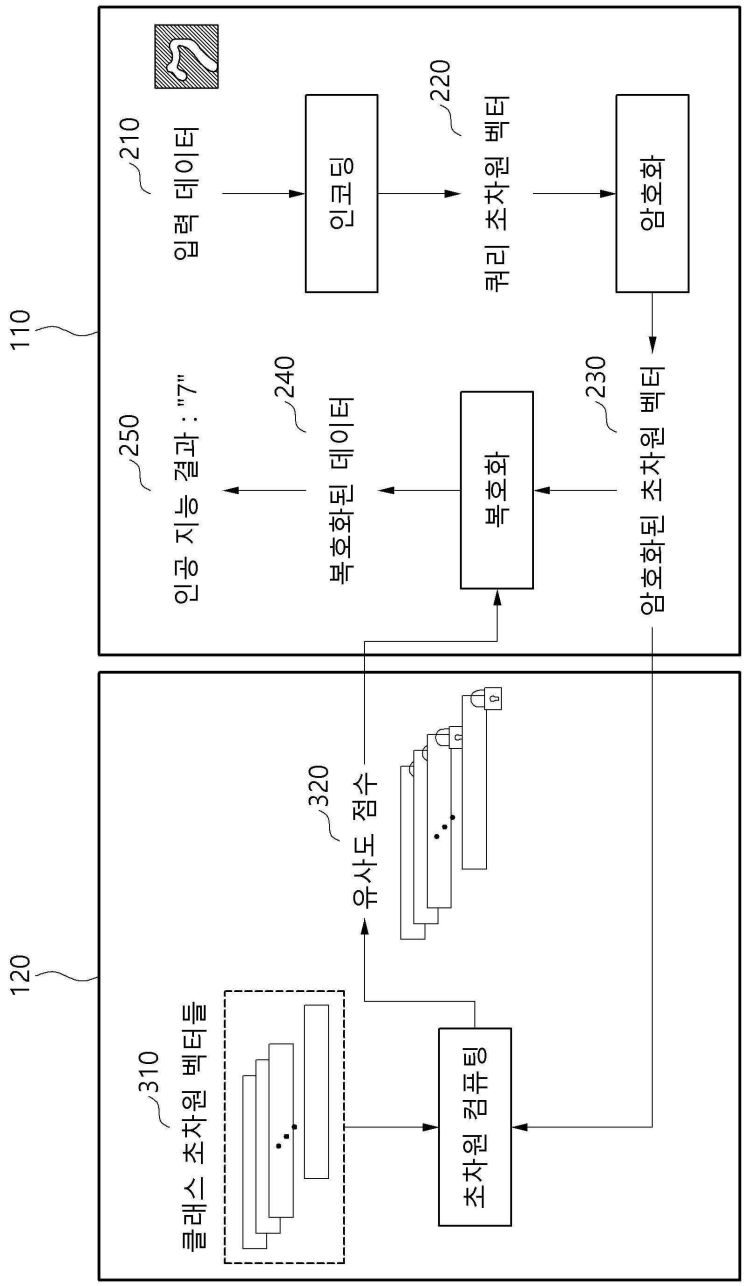
도면1


110
120
암호화된 데이터
클라이언트
서버
연산 결과

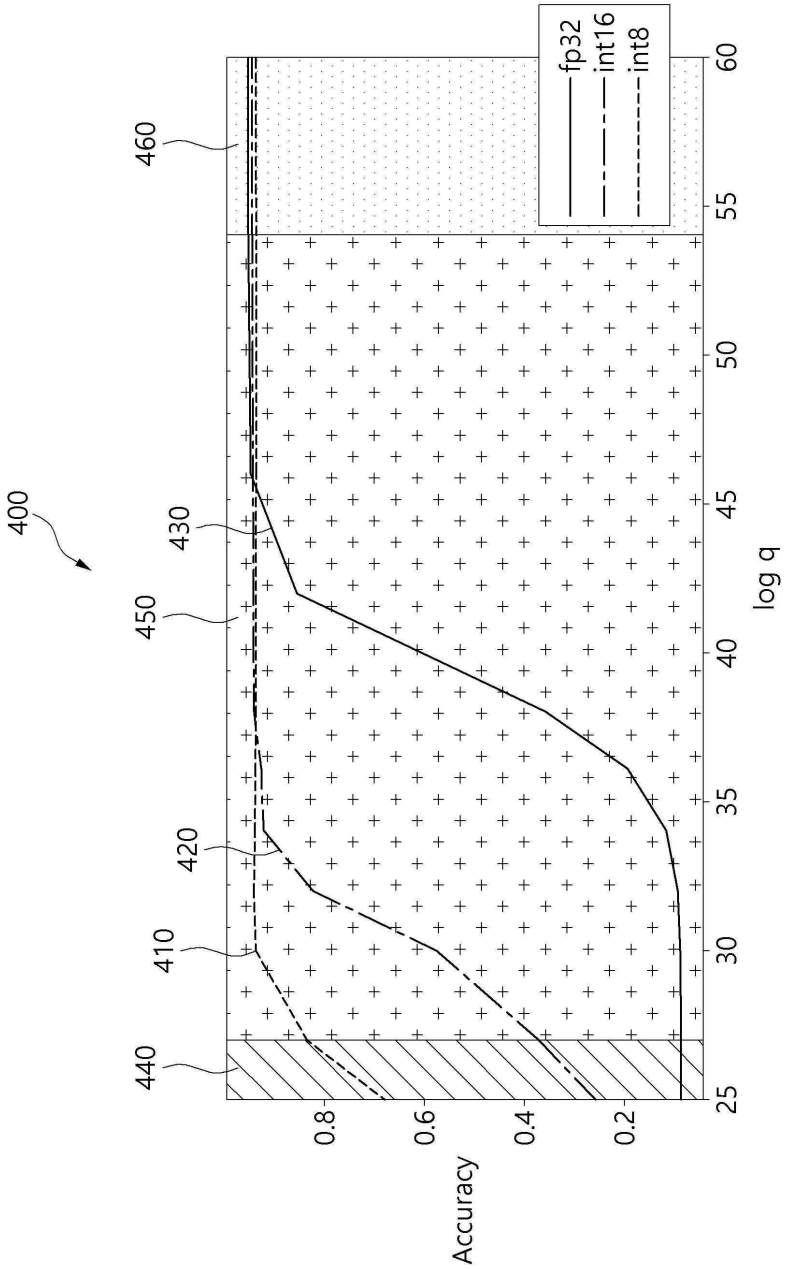
도면2


110
210
입력 데이터
인코딩
220
쿼리 초차원 벡터
암호화
230
암호화된 초차원 벡터
복호화
240
복호화된 데이터
250
인공 지능 결과 : "7"
연산 결과

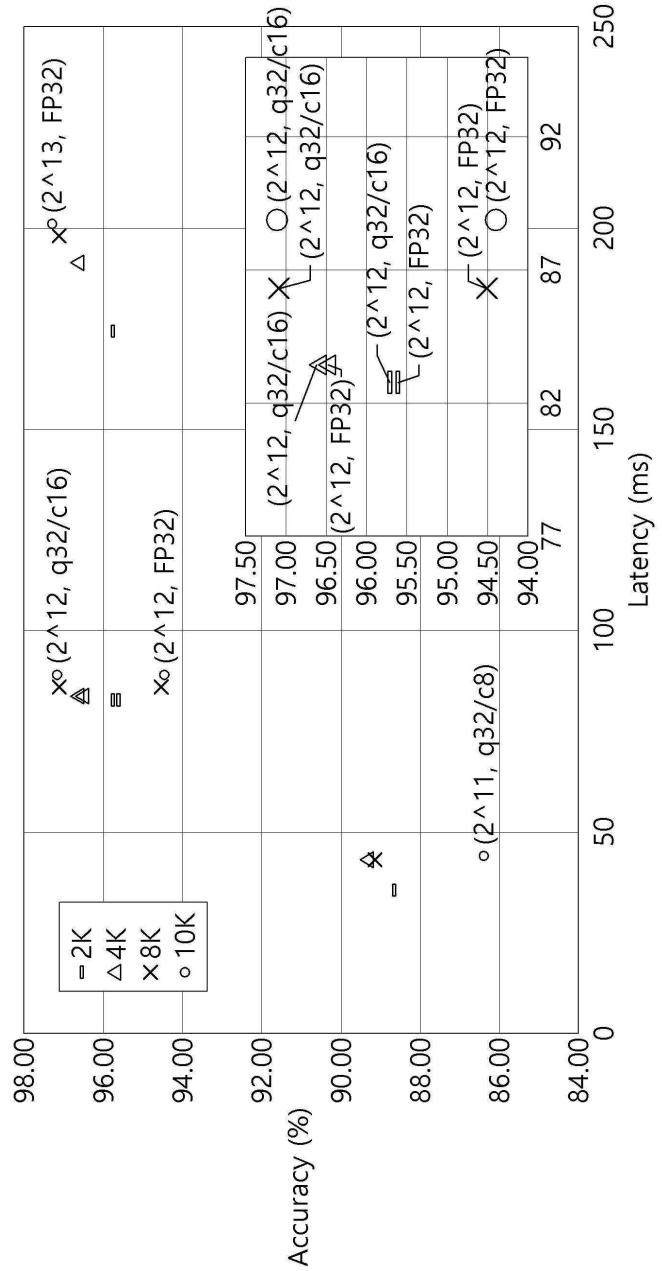
도면3


110
210
입력 데이터
인코딩
220
쿼리 초차원 벡터
암호화
250
인공 지능 결과 : "7"
240
복호화된 데이터
복호화
230
암호화된 초차원 벡터
120
310
클래스 초차원 벡터들
320
유사도 점수
초차원 컴퓨팅

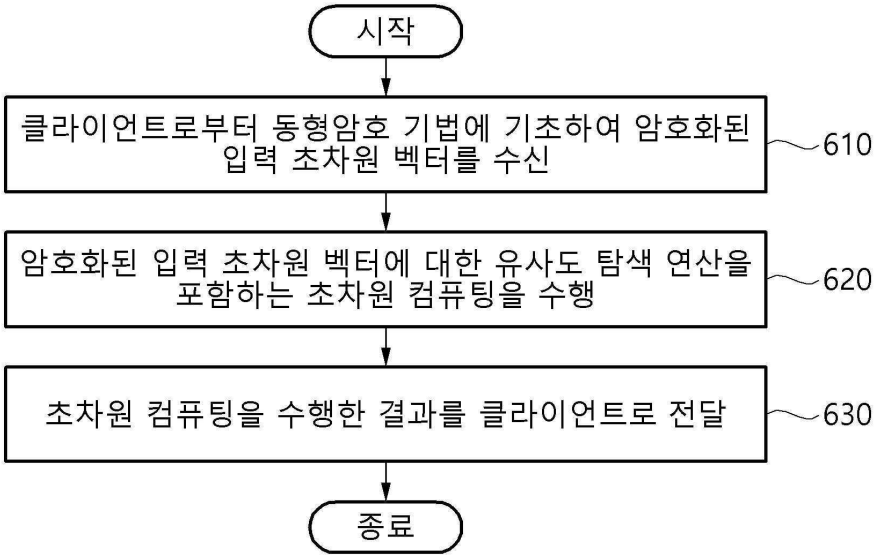
도면4


400
410
420
430
440
450
460
fp32
int16
int8
Accuracy
0.8
0.6
0.4
0.2
log q
60
55
50
45
40
35
30
25

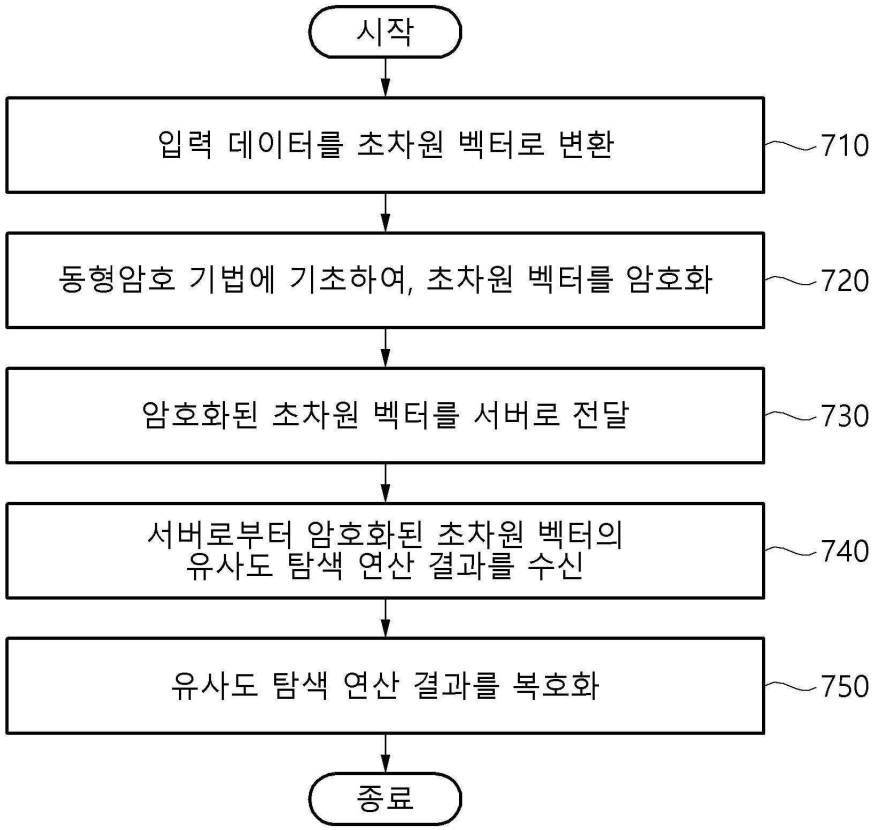
도면5


2K
4K
8K
10K
(2^12, q32/c16)
(2^12, FP32)
(2^13, FP32)
(2^11, q32/c8)
(2^12, q32/c16)
(2^12, FP32)
(2^12, q32/c16)
(2^12, FP32)
(2^12, q32/c16)
(2^12, FP32)
Accuracy (%)
Latency (ms)
98.00
96.00
94.00
92.00
90.00
88.00
86.00
84.00
0
50
100
150
200
250
97.50
97.00
96.50
96.00
95.50
95.00
94.50
94.00
77
82
87
92

도면6


시작
클라이언트로부터 동형암호 기법에 기초하여 암호화된 입력 초차원 벡터를 수신
610
암호화된 입력 초차원 벡터에 대한 유사도 탐색 연산을 포함하는 초차원 컴퓨팅을 수행
620
초차원 컴퓨팅을 수행한 결과를 클라이언트로 전달
630
종료

도면7


시작
입력 데이터를 초차원 벡터로 변환
710
동형암호 기법에 기초하여, 초차원 벡터를 암호화
720
암호화된 초차원 벡터를 서버로 전달
730
서버로부터 암호화된 초차원 벡터의 유사도 탐색 연산 결과를 수신
740
유사도 탐색 연산 결과를 복호화
750
종료