



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0025339
(43) 공개일자 2025년02월21일

- | | |
|---|--|
| <p>(51) 국제특허분류(Int. Cl.)
G06N 3/0475 (2023.01) B64C 39/02 (2023.01)
B64D 45/00 (2025.01) G06N 3/0495 (2023.01)
G06N 3/09 (2023.01)</p> <p>(52) CPC특허분류
G06N 3/0475 (2023.01)
B64C 39/024 (2023.01)</p> <p>(21) 출원번호 10-2024-0108185
(22) 출원일자 2024년08월13일
심사청구일자 2024년08월13일</p> <p>(30) 우선권주장
1020230106440 2023년08월14일 대한민국(KR)</p> | <p>(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)</p> <p>(72) 발명자
김휘강
서울특별시 서초구 서운로 104, 201동 804호(서초동, 래미안서초에스티지에스)</p> <p>유정도
서울특별시 성북구 화랑로5길 4-12 (하월곡동)
(뒷면에 계속)</p> <p>(74) 대리인
이대호, 박건홍</p> |
|---|--|

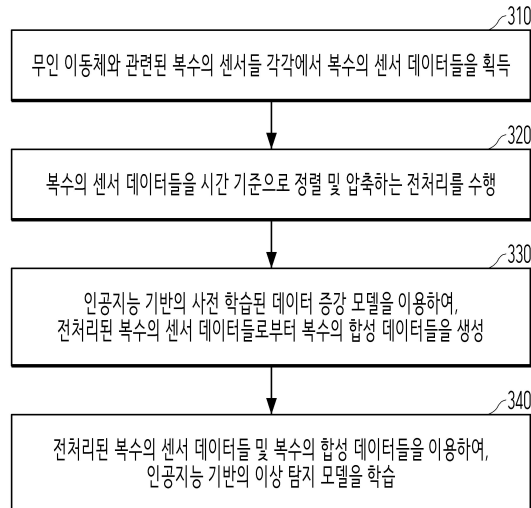
전체 청구항 수 : 총 13 항

(54) 발명의 명칭 무인 이동체의 이상 탐지를 위한 인공지능 모델을 학습시키는 방법 및 장치

(57) 요약

본 개시의 일 실시예에 따라 컴퓨팅 장치에 의해 수행되는, 이상 탐지(anomaly detection)를 위한 모델을 학습시키기 위한 방법으로서, 상기 방법은: 무인 이동체와 관련된 복수의 센서들 각각에서 복수의 센서 데이터들을 획득하는 단계; 상기 복수의 센서 데이터들을 시간 기준으로 정렬 및 압축하는 전처리를 수행하는 단계; 인공지능 기반의 사전 학습된 데이터 증강 모델을 이용하여, 전처리된 상기 복수의 센서 데이터들로부터 복수의 합성 데이터들을 생성하는 단계; 및 전처리된 상기 복수의 센서 데이터들 및 상기 복수의 합성 데이터들을 이용하여, 인공지능 기반의 이상 탐지 모델을 학습시키는 단계;를 포함할 수 있다.

대표도 - 도3



(52) CPC특허분류

B64D 45/0031 (2019.08)

G06N 3/0495 (2023.01)

G06N 3/09 (2023.01)

(72) 발명자

송민근

서울특별시 동대문구 무학로49길 22 (용두동)

김동영

서울특별시 성북구 고려대로24길 60 (안암동5가)

송유찬

서울특별시 성북구 안암로9가길 87-3 (안암동5가)

김강민

서울특별시 동대문구 무학로 193 (용두동)

김동성

서울특별시 성북구 안암로9가길 72, 105호(안암동5가)

이 발명을 지원한 국가연구개발사업

과제고유번호

미부여

과제번호

2020-0-00374-004

부처명

과학기술정보통신부

과제관리(전문)기관명

정보통신기획평가원

연구사업명

ICT인프라·서비스보호강화

연구과제명

무인이동체 공통핵심 보안기술 개발 연구

과제수행기관명

고려대학교 산학협력단

연구기간

2020.04.01 ~ 2027.12.31

명세서

청구범위

청구항 1

컴퓨팅 장치에 의해 수행되는, 이상 탐지(anomaly detection)를 위한 모델을 학습시키기 위한 방법으로서, 상기 방법은:

무인 이동체와 관련된 복수의 센서들 각각에서 복수의 센서 데이터들을 획득하는 단계;

상기 복수의 센서 데이터들을 시간 기준으로 정렬 및 압축하는 전처리를 수행하는 단계;

인공지능 기반의 사전 학습된 데이터 증강 모델을 이용하여, 전처리된 상기 복수의 센서 데이터들로부터 복수의 합성 데이터들을 생성하는 단계; 및

전처리된 상기 복수의 센서 데이터들 및 상기 복수의 합성 데이터들을 이용하여, 인공지능 기반의 이상 탐지 모델을 학습시키는 단계;

를 포함하는,

방법.

청구항 2

제1항에 있어서,

상기 전처리를 수행하는 단계는,

상기 복수의 센서 데이터들을 사전 결정된 시간 구간에 따라 분류하는 단계;

상기 사전 결정된 시간 구간별로 대응되는 센서 데이터의 변화량을 산출하는 단계;

상기 사전 결정된 시간 구간별로 대응되는 상기 센서 데이터를 압축하는 단계; 및

사전 결정된 특징 선택 알고리즘을 이용하여, 상기 센서 데이터의 변화량 및 압축된 상기 센서 데이터로부터 적어도 하나의 특징(feature)을 선정하는 단계;

를 포함하는,

방법.

청구항 3

제2항에 있어서,

상기 센서 데이터의 변화량을 산출하는 단계는,

제1 시간 구간의 제1 세트의 센서 데이터들 중에서, 수치형(numerical) 센서 데이터를 생성하는 제1 센서로부터 획득된 제1 센서 데이터의 제1 변화량을 산출하는 단계;

를 포함하는,

방법.

청구항 4

제3항에 있어서,

상기 제1 변화량은,

상기 제1 시간 구간보다 이전의 시간 구간에서 상기 제1 센서로부터 획득된 이전 센서 데이터와, 상기 제1 센서 데이터 간의 차이 값을 포함하는,

방법.

청구항 5

제3항에 있어서,

상기 제1 변화량은,

상기 제1 센서 데이터가 복수인 경우, 복수의 제1 센서 데이터들 간의 차이 값을 포함하는,

방법.

청구항 6

제3항에 있어서,

상기 센서 데이터를 압축하는 단계는,

상기 제1 세트의 센서 데이터들의 주기를 통일하기 위해서, 상기 제1 세트의 센서 데이터들을 각각 사전 결정된 헤르츠(Hz)로 압축하는 단계;

를 포함하는,

방법.

청구항 7

제6항에 있어서,

상기 제1 세트의 센서 데이터들을 각각 사전 결정된 헤르츠(Hz)로 압축하는 단계는,

상기 제1 세트의 센서 데이터들 중에서 센서별로 각각 하나의 대표 센서 데이터를 결정하는 단계; 및

상기 센서별로 각각 결정된 상기 대표 센서 데이터를 상기 사전 결정된 헤르츠로 압축하는 단계;

를 포함하는,

방법.

청구항 8

제7항에 있어서,

상기 제1 세트의 센서 데이터들 중에서 센서별로 각각 하나의 대표 센서 데이터를 결정하는 단계는,

수치형 센서 데이터를 생성하는 제2 센서에서 획득된 복수의 제2 센서 데이터들 중에서, 상기 복수의 제2 센서 데이터들의 평균 값에 대응되는 제2-1 센서 데이터 또는 상기 평균 값을 상기 제2 센서의 대표 센서 데이터로 결정하는 단계; 및

범주형(categorical) 센서 데이터를 생성하는 제3 센서에서 획득된 복수의 제3 센서 데이터들 중에서, 복수의 제3 센서 데이터들의 최빈값(mode)에 대응되는 제3-1 센서 데이터 또는 상기 최빈값을 상기 제3 센서의 대표 센서 데이터로 결정하는 단계;

를 포함하는,

방법.

청구항 9

제2항에 있어서,

상기 적어도 하나의 특징을 선정하는 단계는,

상기 사전 결정된 특징 선택 알고리즘이 복수인 경우, 그레디언트 부스팅(gradient boosting) 알고리즘을 포함하는 제1 특징 선택 알고리즘을 이용하여, 상기 센서 데이터의 변화량 및 압축된 상기 센서 데이터로부터 적어도 하나의 제1 특징을 선정하는 단계;

로지스틱 회귀(logistic regression) 알고리즘을 포함하는 제2 특징 선택 알고리즘을 이용하여, 상기 센서 데이터의 변화량 및 압축된 상기 센서 데이터로부터 적어도 하나의 제2 특징을 선정하는 단계; 및

상기 적어도 하나의 제1 특징과 상기 적어도 하나의 제2 특징을 비교함으로써, 서로 중복되는 특징인 적어도 하나의 공통 특징을 선정하는 단계;

를 포함하는,

방법.

청구항 10

제1항에 있어서,

상기 데이터 증강 모델은,

전처리된 상기 복수의 센서 데이터들의 각각에 정규 분포로부터 생성된 노이즈를 추가하는 확산 프로세스(diffusion process)를 수행하고 그리고 추가된 상기 노이즈를 제거하는 역확산 프로세스(reverse diffusion process)를 수행함으로써, 전처리된 상기 복수의 센서 데이터들과 유사한 상기 복수의 합성 데이터들을 생성하는,

방법.

청구항 11

제1항에 있어서,

상기 이상 탐지 모델은,

전처리된 상기 복수의 센서 데이터들 및 상기 복수의 합성 데이터들을 포함하는 학습 데이터와, 상기 학습 데이터를 입력 받음으로써 출력되는 출력 데이터 간의 손실 값을 줄이는 방향으로 학습되는,

방법.

청구항 12

컴퓨터 판독가능 저장 매체에 저장된 컴퓨터 프로그램으로서, 상기 컴퓨터 프로그램은 컴퓨팅 장치의 프로세서로 하여금 이상 탐지(anomaly detection)를 위한 모델을 학습시키기 위한 방법을 수행하도록 하며, 상기 방법은:

무인 이동체와 관련된 복수의 센서들 각각에서 복수의 센서 데이터들을 획득하는 단계;

상기 복수의 센서 데이터들을 시간 기준으로 정렬 및 압축하는 전처리를 수행하는 단계;

인공지능 기반의 사전 학습된 데이터 증강 모델을 이용하여, 전처리된 상기 복수의 센서 데이터들로부터 복수의 합성 데이터들을 생성하는 단계; 및

전처리된 상기 복수의 센서 데이터들 및 상기 복수의 합성 데이터들을 이용하여, 인공지능 기반의 이상 탐지 모델을 학습시키는 단계;

를 포함하는,

컴퓨터 판독가능 저장 매체에 저장된 컴퓨터 프로그램.

청구항 13

이상 탐지(anomaly detection)를 위한 모델을 학습시키기 위한 컴퓨팅 장치에 있어서,

프로세서;

메모리; 및

네트워크부;

를 포함하고,

상기 프로세서는:

무인 이동체와 관련된 복수의 센서들 각각에서 복수의 센서 데이터들을 획득하는 동작;

상기 복수의 센서 데이터들을 시간 기준으로 정렬 및 압축하는 전처리를 수행하는 동작;

인공지능 기반의 사전 학습된 데이터 증강 모델을 이용하여, 전처리된 상기 복수의 센서 데이터들로부터 복수의 합성 데이터들을 생성하는 동작; 및

전처리된 상기 복수의 센서 데이터들 및 상기 복수의 합성 데이터들을 이용하여, 인공지능 기반의 이상 탐지 모델을 학습시키는 동작;

을 수행하는,

컴퓨팅 장치.

발명의 설명

기술 분야

[0001] 본 개시는 이상 탐지에 관한 것으로, 보다 구체적으로, 무인 이동체의 이상 탐지를 위한 인공지능 모델을 학습시키는 방법 및 장치에 관한 것이다.

배경 기술

[0002] 무인 이동체는 사람의 직접적인 조종 없이 운행하는 이동체를 의미할 수 있다. 이러한 무인 이동체는 운행되는 환경에 따라, 공중에서 운용되는 UAV(Unmanned Aerial Vehicle), 지상에서 운용되는 UGV(Unmanned Ground Vehicle), 해상에서 운용되는 USV(Unmanned Surface Vehicle), 수중에서 운용되는 AUV (Autonomous Underwater Vehicle)로 구분될 수 있다.

[0003] 무인 이동체는 인터넷 등의 네트워크와 연결되며, 센서를 통해 사람의 개입 없이 자동으로 판단할 수도 있다. 따라서, 이상이 발생할 경우, 사람이 즉각적으로 개입할 수 없는 경우가 발생할 수 있다. 이러한 경우를 방지하기 위해, 무인 이동체에 대해 자동으로 보안 위험을 탐지할 수 있는 이상 탐지 모델의 필요성이 대두되고 있다.

선행기술문헌

특허문헌

[0004] (특허문헌 0001) 대한민국 등록특허 제10-2417518호(2022.07.01. 등록)

발명의 내용

해결하려는 과제

[0005] 본 개시는 전술한 배경기술에 대응하여 안출된 것으로, 무인 이동체의 이상 탐지를 정확하게 수행하기 위한 인공지능 모델을 학습시키는 방법 및 장치를 제공하고자 한다.

[0006] 본 개시의 기술적 과제들은 이상에서 언급한 기술적 과제로 제한되지 않으며, 언급되지 않은 또 다른 기술적 과제들은 아래의 기재로부터 당업자에게 명확하게 이해될 수 있을 것이다.

과제의 해결 수단

[0007] 전술한 바와 같은 과제를 해결하기 위한 본 개시의 일 실시예에 따라, 컴퓨팅 장치에 의해 수행되는, 이상 탐지(anomaly detection)를 위한 모델을 학습시키기 위한 방법이 개시된다. 상기 방법은: 무인 이동체와 관련된 복수의 센서들 각각에서 복수의 센서 데이터들을 획득하는 단계; 상기 복수의 센서 데이터들을 시간 기준으로 정렬 및 압축하는 전처리를 수행하는 단계; 인공지능 기반의 사전 학습된 데이터 증강 모델을 이용하여, 전처리된 상기 복수의 센서 데이터들로부터 복수의 합성 데이터들을 생성하는 단계; 및 전처리된 상기 복수의 센서 데이터들 및 상기 복수의 합성 데이터들을 이용하여, 인공지능 기반의 이상 탐지 모델을 학습시키는 단계;를 포함할

수 있다.

- [0008] 일 실시예에서, 상기 전처리를 수행하는 단계는, 상기 복수의 센서 데이터들을 사전 결정된 시간 구간에 따라 분류하는 단계; 상기 사전 결정된 시간 구간별로 대응되는 센서 데이터의 변화량을 산출하는 단계; 상기 사전 결정된 시간 구간별로 대응되는 상기 센서 데이터를 압축하는 단계; 및 사전 결정된 특징 선택 알고리즘을 이용하여, 상기 센서 데이터의 변화량 및 압축된 상기 센서 데이터로부터 적어도 하나의 특징(feature)을 선정하는 단계;를 포함할 수 있다.
- [0009] 일 실시예에서, 상기 센서 데이터의 변화량을 산출하는 단계는, 제1 시간 구간의 제1 세트의 센서 데이터들 중에서, 수치형(numerical) 센서 데이터를 생성하는 제1 센서로부터 획득된 제1 센서 데이터의 제1 변화량을 산출하는 단계;를 포함할 수 있다.
- [0010] 일 실시예에서, 상기 제1 변화량은, 상기 제1 시간 구간보다 이전의 시간 구간에서 상기 제1 센서로부터 획득된 이전 센서 데이터와, 상기 제1 센서 데이터 간의 차이 값을 포함할 수 있다.
- [0011] 일 실시예에서, 상기 제1 변화량은, 상기 제1 센서 데이터가 복수인 경우, 복수의 제1 센서 데이터들 간의 차이 값을 포함할 수 있다.
- [0012] 일 실시예에서, 상기 센서 데이터를 압축하는 단계는, 상기 제1 세트의 센서 데이터들의 주기를 통일하기 위해서, 상기 제1 세트의 센서 데이터들을 각각 사전 결정된 헤르츠(Hz)로 압축하는 단계;를 포함할 수 있다.
- [0013] 일 실시예에서, 상기 제1 세트의 센서 데이터들을 각각 사전 결정된 헤르츠(Hz)로 압축하는 단계는, 상기 제1 세트의 센서 데이터들 중에서 센서별로 각각 하나의 대표 센서 데이터를 결정하는 단계; 및 상기 센서별로 각각 결정된 상기 대표 센서 데이터를 상기 사전 결정된 헤르츠로 압축하는 단계;를 포함할 수 있다.
- [0014] 일 실시예에서, 상기 제1 세트의 센서 데이터들 중에서 센서별로 각각 하나의 대표 센서 데이터를 결정하는 단계는, 수치형 센서 데이터를 생성하는 제2 센서에서 획득된 복수의 제2 센서 데이터들 중에서, 상기 복수의 제2 센서 데이터들의 평균 값에 대응되는 제2-1 센서 데이터 또는 상기 평균 값을 상기 제2 센서의 대표 센서 데이터로 결정하는 단계; 및 범주형(categorical) 센서 데이터를 생성하는 제3 센서에서 획득된 복수의 제3 센서 데이터들 중에서, 복수의 제3 센서 데이터들의 최빈값(mode)에 대응되는 제3-1 센서 데이터 또는 상기 최빈값을 상기 제3 센서의 대표 센서 데이터로 결정하는 단계;를 포함할 수 있다.
- [0015] 일 실시예에서, 상기 적어도 하나의 특징을 선정하는 단계는, 상기 사전 결정된 특징 선택 알고리즘이 복수인 경우, 그래디언트 부스팅(gradient boosting) 알고리즘을 포함하는 제1 특징 선택 알고리즘을 이용하여, 상기 센서 데이터의 변화량 및 압축된 상기 센서 데이터로부터 적어도 하나의 제1 특징을 선정하는 단계; 로지스틱 회귀(logistic regression) 알고리즘을 포함하는 제2 특징 선택 알고리즘을 이용하여, 상기 센서 데이터의 변화량 및 압축된 상기 센서 데이터로부터 적어도 하나의 제2 특징을 선정하는 단계; 및 상기 적어도 하나의 제1 특징과 상기 적어도 하나의 제2 특징을 비교함으로써, 서로 중복되는 특징인 적어도 하나의 공통 특징을 선정하는 단계;를 포함할 수 있다.
- [0016] 일 실시예에서, 상기 데이터 증강 모델은, 전처리된 상기 복수의 센서 데이터들의 각각에 정규 분포로부터 생성된 노이즈를 추가하는 확산 프로세스(diffusion process)를 수행하고 그리고 추가된 상기 노이즈를 제거하는 역 확산 프로세스(reverse diffusion process)를 수행함으로써, 전처리된 상기 복수의 센서 데이터들과 유사한 상기 복수의 합성 데이터들을 생성할 수 있다.
- [0017] 일 실시예에서, 상기 이상 탐지 모델은, 전처리된 상기 복수의 센서 데이터들 및 상기 복수의 합성 데이터들을 포함하는 학습 데이터와, 상기 학습 데이터를 입력 받음으로써 출력되는 출력 데이터 간의 손실 값을 줄이는 방향으로 학습될 수 있다.
- [0018] 일 실시예에서, 컴퓨터 판독가능 저장 매체에 저장된 컴퓨터 프로그램이 개시된다. 상기 컴퓨터 프로그램은 컴퓨팅 장치의 프로세서로 하여금 이상 탐지(anomaly detection)를 위한 모델을 학습시키기 위한 방법을 수행하도록 한다. 상기 방법은: 무인 이동체와 관련된 복수의 센서들 각각에서 복수의 센서 데이터들을 획득하는 단계; 상기 복수의 센서 데이터들을 시간 기준으로 정렬 및 압축하는 전처리를 수행하는 단계; 인공지능 기반의 사전 학습된 데이터 증강 모델을 이용하여, 전처리된 상기 복수의 센서 데이터들로부터 복수의 합성 데이터들을 생성하는 단계; 및 전처리된 상기 복수의 센서 데이터들 및 상기 복수의 합성 데이터들을 이용하여, 인공지능 기반의 이상 탐지 모델을 학습시키는 단계;를 포함할 수 있다.
- [0019] 일 실시예에서, 이상 탐지(anomaly detection)를 위한 모델을 학습시키기 위한 컴퓨팅 장치가 개시된다. 상기

컴퓨팅 장치는 프로세서; 메모리; 및 네트워크부;를 포함할 수 있다. 상기 프로세서는: 무인 이동체와 관련된 복수의 센서들 각각에서 복수의 센서 데이터들을 획득하는 동작; 상기 복수의 센서 데이터들을 시간 기준으로 정렬 및 압축하는 전처리를 수행하는 동작; 인공지능 기반의 사전 학습된 데이터 증강 모델을 이용하여, 전처리된 상기 복수의 센서 데이터들로부터 복수의 합성 데이터들을 생성하는 동작; 및 전처리된 상기 복수의 센서 데이터들 및 상기 복수의 합성 데이터들을 이용하여, 인공지능 기반의 이상 탐지 모델을 학습시키는 동작;을 수행할 수 있다.

발명의 효과

- [0020] 본 개시는 무인 이동체의 이상 탐지를 정확하게 수행하기 위한 인공지능 모델을 학습시키는 방법 및 장치를 제공한다.
- [0021] 본 개시는 무인 이동체에 대한 이상 탐지를 위해 무인 이동체 관련 센서 데이터를 전처리함으로써, 인공지능 모델의 학습이 더 원활하게 이루어지도록 할 수 있다.
- [0022] 본 개시는 데이터 증강 모델을 이용하여 다양한 패턴을 지닌 데이터를 생성하여, 이상 탐지 모델의 성능을 더욱 높이고, 무인 이동체와 관련된 데이터를 생성하는 비용을 줄이는 효과가 있다.
- [0023] 본 개시에서 얻을 수 있는 효과는 이상에서 언급한 효과로 제한되지 않으며, 언급하지 않은 또 다른 효과들은 아래의 기재로부터 본 개시가 속하는 기술분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있을 것이다.

도면의 간단한 설명

- [0024] 다양한 양상들이 이제 도면들을 참조로 기재되며, 여기서 유사한 참조 번호들은 총괄적으로 유사한 구성요소들을 지칭하는데 이용된다. 이하의 실시예에서, 설명 목적을 위해, 다수의 특정 세부사항들이 하나 이상의 양상들의 총체적 이해를 제공하기 위해 제시된다. 그러나, 그러한 양상(들)이 이러한 구체적인 세부사항들 없이 실시될 수 있음은 명백할 것이다.
- 도 1은 본 개시의 일 실시예에 따른 이상 탐지를 위한 모델을 학습시키는 컴퓨팅 장치를 설명하기 위한 도면이다.
- 도 2는 본 개시의 일 실시예에 따른 인공지능 모델의 예시적인 구조를 나타낸 도면이다.
- 도 3은 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 이상 탐지를 위한 모델을 학습시키기 위한 방법을 나타낸 도면이다.
- 도 4는 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 이상 탐지를 위한 모델의 학습에 사용되는 데이터의 전처리 방법을 나타낸 도면이다.
- 도 5는 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 센서별로 대표 센서 데이터를 결정하는 과정을 나타낸 도면이다.
- 도 6은 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 데이터 증강 모델을 이용하여 합성 데이터를 생성하는 과정을 나타낸 도면이다.
- 도 7은 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 이상 탐지 모델을 학습시키는 과정을 나타낸 도면이다.
- 도 8은 본 개시의 실시예들이 구현될 수 있는 예시적인 컴퓨팅 환경에 대한 간략하고 일반적인 개략도를 도시한다.

발명을 실시하기 위한 구체적인 내용

- [0025] 다양한 실시예들이 이제 도면을 참조하여 설명된다. 본 명세서에서, 다양한 설명들이 본 개시의 이해를 제공하기 위해서 제시된다. 그러나, 이러한 실시예들은 이러한 구체적인 설명 없이도 실행될 수 있음이 명백하다.
- [0026] 본 명세서에서 사용되는 용어 "컴포넌트", "모듈", "시스템" 등은 컴퓨터-관련 엔티티, 하드웨어, 펌웨어, 소프트웨어, 소프트웨어 및 하드웨어의 조합, 또는 소프트웨어의 실행을 지칭한다. 예를 들어, 컴포넌트는 프로세서 상에서 실행되는 처리과정(procedure), 프로세서, 객체, 실행 스레드, 프로그램, 및/또는 컴퓨터일 수 있지만,

이들로 제한되는 것은 아니다. 예를 들어, 컴퓨팅 장치에서 실행되는 애플리케이션 및 컴퓨팅 장치 모두 컴포넌트일 수 있다. 하나 이상의 컴포넌트는 프로세서 및/또는 실행 스레드 내에 상주할 수 있다. 일 컴포넌트는 하나의 컴퓨터 내에 로컬화 될 수 있다. 일 컴포넌트는 2개 이상의 컴퓨터들 사이에 분배될 수 있다. 또한, 이러한 컴포넌트들은 그 내부에 저장된 다양한 데이터 구조들을 갖는 다양한 컴퓨터 관독가능한 매체로부터 실행할 수 있다. 컴포넌트들은 예를 들어 하나 이상의 데이터 패킷들을 갖는 신호(예를 들면, 로컬 시스템, 분산 시스템에서 다른 컴포넌트와 상호작용하는 하나의 컴포넌트로부터의 데이터 및/또는 신호를 통해 다른 시스템과 인터넷과 같은 네트워크를 통해 전송되는 데이터)에 따라 로컬 및/또는 원격 처리들을 통해 통신할 수 있다.

[0027] 더불어, 용어 "또는"은 배타적 "또는"이 아니라 내포적 "또는"을 의미하는 것으로 의도된다. 즉, 달리 특정되지 않거나 문맥상 명확하지 않은 경우에, "X는 A 또는 B를 이용한다"는 자연적인 내포적 치환 중 하나를 의미하는 것으로 의도된다. 즉, X가 A를 이용하거나; X가 B를 이용하거나; 또는 X가 A 및 B 모두를 이용하는 경우, "X는 A 또는 B를 이용한다"가 이들 경우들 어느 것으로도 적용될 수 있다. 또한, 본 명세서에 사용된 "또는", "및/또는"이라는 용어는 열거된 관련 아이템들 중 하나 이상의 아이템의 가능한 모든 조합을 지칭하고 포함하는 것으로 이해되어야 한다.

[0028] 또한, "포함한다" 및/또는 "포함하는"이라는 용어는, 해당 특징 및/또는 구성요소가 존재함을 의미하는 것으로 이해되어야 한다. 다만, "포함한다" 및/또는 "포함하는"이라는 용어는, 하나 이상의 다른 특징, 구성요소 및/또는 이들의 그룹의 존재 또는 추가를 배제하지 않는 것으로 이해되어야 한다. 또한, 달리 특정되지 않거나 단수 형태를 지시하는 것으로 문맥상 명확하지 않은 경우에, 본 명세서와 청구범위에서 단수는 일반적으로 "하나 또는 그 이상"을 의미하는 것으로 해석되어야 한다.

[0029] 그리고, "A 또는 B 중 적어도 하나"이라는 용어는, "A만을 포함하는 경우", "B 만을 포함하는 경우", "A와 B의 구성으로 조합된 경우"를 의미하는 것으로 해석되어야 한다.

[0030] 당업자들은 추가적으로 여기서 개시된 실시예들과 관련되어 설명된 다양한 예시적 논리적 블록들, 구성들, 모듈들, 회로들, 수단들, 로직들, 및 알고리즘 단계들이 전자 하드웨어, 컴퓨터 소프트웨어, 또는 양쪽 모두의 조합들로 구현될 수 있음을 인식해야 한다. 하드웨어 및 소프트웨어의 상호교환성을 명백하게 예시하기 위해, 다양한 예시적 컴포넌트들, 블록들, 구성들, 수단들, 로직들, 모듈들, 회로들, 및 단계들은 그들의 기능성 측면에서 일반적으로 위에서 설명되었다. 그러한 기능성이 하드웨어로 또는 소프트웨어로서 구현되는지 여부는 전반적인 시스템에 부과된 특정 어플리케이션(application) 및 설계 제한들에 달려 있다. 숙련된 기술자들은 각각의 특정 어플리케이션들을 위해 다양한 방법으로 설명된 기능성을 구현할 수 있다. 다만, 그러한 구현의 결정들이 본 개시내용의 영역을 벗어나게 하는 것으로 해석되어서는 안 된다.

[0031] 제시된 실시예들에 대한 설명은 본 개시의 기술 분야에서 통상의 지식을 가진 자가 본 발명을 이용하거나 또는 실시할 수 있도록 제공된다. 이러한 실시예들에 대한 다양한 변형들은 본 개시의 기술 분야에서 통상의 지식을 가진 자에게 명백할 것이다. 여기에 정의된 일반적인 원리들은 본 개시의 범위를 벗어나지 않고 다른 실시예들에 적용될 수 있다. 그리하여, 본 개시는 여기에 제시된 실시예 들로 한정되는 것이 아니다. 본 개시는 여기에 제시된 원리들 및 신규한 특징들과 일관되는 최광의의 범위에서 해석되어야 할 것이다.

[0032] 본 개시내용에서의 제1, 제2, 또는 제3 과 같이 제 N으로 표현되는 용어들은 적어도 하나의 엔티티들을 구분하기 위해 사용된다. 예를 들어, 제1 과 제2 로 표현된 엔티티들은 서로 동일하거나 또는 상이할 수 있다.

[0033] 그리고, "A, B 등"과 같이 "~ 등"이라는 용어는, "A만을 포함하는 경우", "B 만을 포함하는 경우", "A와 B의 구성으로 조합된 경우"를 의미하는 것으로 해석되어야 한다.

[0034] 도 1은 본 개시의 일 실시예에 따른 이상 탐지를 위한 모델을 학습시키는 컴퓨팅 장치를 설명하기 위한 도면이다.

[0035] 도 1을 참조하면, 이상 탐지를 위한 인공지능 모델을 학습시키는 컴퓨팅 장치(100)는 프로세서(110), 메모리(130), 네트워크부(150)를 포함할 수 있다. 상술한 컴퓨팅 장치(100)의 구성은 간략화 하여 나타낸 예시일 뿐이다. 일 실시예에서, 컴퓨팅 장치(100)는 임의의 형태의 서버, 사용자 디바이스 등을 포함할 수 있다.

[0036] 프로세서(110)는 하나 이상의 코어로 구성될 수 있으며, 컴퓨팅 장치(100)의 중앙 처리 장치(CPU: central processing unit), 범용 그래픽 처리 장치(GPGPU: general purpose graphics processing unit), 텐서 처리 장치(TPU: tensor processing unit)등의 데이터의 처리에 관련된 동작을 수행하기 위한 프로세서를 포함할 수 있다.

- [0037] 본 개시의 일 실시예에 따라 프로세서(110)는 신경망의 학습을 위한 연산을 수행할 수도 있다. 예를 들어, 프로세서(110)는 딥러닝(DL: deep learning)에서 학습을 위한 입력 데이터의 처리, 입력 데이터에서의 피쳐 추출, 오차 계산, 역전파(backpropagation)를 이용한 신경망의 가중치 업데이트 등의 신경망의 학습을 위한 계산을 수행할 수 있다. 프로세서(110)의 CPU, GPGPU, 및 TPU 중 적어도 하나가 네트워크 함수의 학습을 처리할 수 있다. 예를 들어, CPU 와 GPGPU가 함께 네트워크 함수의 학습, 네트워크 함수를 이용한 데이터 분류를 처리할 수 있다. 또한, 본 개시의 일 실시예에서 복수의 컴퓨팅 장치들의 프로세서들을 함께 사용하여 네트워크 함수의 학습, 네트워크 함수를 이용한 데이터 분류를 처리할 수도 있다. 예를 들어, 프로세서(110)는 이상 탐지를 위한 인공지능 기반의 이상 탐지 모델을 학습시킬 수 있다. 프로세서(110)는 학습된 이상 탐지 모델에 특정 센서 데이터를 입력함으로써, 학습된 이상 탐지 모델에서 특정 센서 데이터의 이상 유무 및/또는 특정 센서 데이터에 대응되는 무인 이동체의 이상 유무가 출력되도록 할 수 있다.
- [0038] 프로세서(110)는 통상적으로 컴퓨팅 장치(100)의 전반적인 동작을 제어할 수 있다. 프로세서(110)는 컴퓨팅 장치(100)에 포함된 구성요소들을 통해 입력 또는 출력되는 신호, 데이터, 정보 등을 처리하거나 메모리(130)에 저장된 응용 프로그램을 구동함으로써, 사용자에게 적절한 정보 또는 기능을 제공 또는 처리할 수 있다.
- [0039] 본 개시의 일 실시예에서, 메모리(130)는 프로세서(110)가 생성하거나 결정한 임의의 형태의 정보 및/또는 네트워크부(150)가 수신한 임의의 형태의 정보를 저장할 수 있다. 일 실시예에서, 메모리(130)는 데이터베이스를 저장하고 있을 수 있다. 데이터베이스는 컴퓨팅 장치(100)가 처리할 수 있는 형태로 저장된 데이터의 집합일 수 있다.
- [0040] 본 개시의 일 실시예에서, 메모리(130)는 플래시 메모리 타입(flash memory type), 하드디스크 타입(hard disk type), 멀티미디어 카드 마이크로 타입(multimedia card micro type), 카드 타입의 메모리(예를 들어 SD 또는 XD 메모리 등), 램(Random Access Memory, RAM), SRAM(Static Random Access Memory), 롬(Read-Only Memory, ROM), EEPROM(Electrically Erasable Programmable Read-Only Memory), PROM(Programmable Read-Only Memory), 자기 메모리, 자기 디스크 및/또는 광디스크 중 적어도 하나의 타입의 저장매체를 포함할 수 있다. 컴퓨팅 장치(100)는 인터넷(internet) 상에서 메모리(130)의 저장 기능을 수행하는 웹 스토리지(web storage)와 관련되어 동작할 수도 있다. 전술한 메모리에 대한 기재는 예시일 뿐, 본 개시는 이에 제한되지 않는다. 메모리(130)는 프로세서(110)에 의하여 동작 될 수 있다.
- [0041] 본 개시의 일 실시예에 따른 네트워크부(150)는 임의의 형태의 데이터 및 신호 등을 송신 및 수신할 수 있는 임의의 유무선 통신 네트워크가 본 개시 내용에서 표현되는 네트워크에 포함될 수 있다. 본 명세서에서 설명된 기술들은 위에서 언급된 네트워크들뿐만 아니라, 다른 네트워크들에서도 사용될 수 있다.
- [0042] 도 2는 본 개시의 일 실시예에 따른 인공지능 모델의 예시적인 구조를 나타낸 도면이다.
- [0043] 본 명세서에 걸쳐, 인공지능 모델, 인공지능 기반의 모델, 연산 모델, 신경망, 네트워크 함수, 뉴럴 네트워크(neural network)는 동일한 의미로 사용될 수 있다.
- [0044] 신경망은 일반적으로 노드라 지칭될 수 있는 상호 연결된 계산 단위들의 집합으로 구성될 수 있다. 이러한 노드들은 뉴런(neuron)들로 지칭될 수도 있다. 신경망은 적어도 하나 이상의 노드들을 포함하여 구성된다. 신경망들을 구성하는 노드(또는 뉴런)들은 하나 이상의 링크에 의해 상호 연결될 수 있다.
- [0045] 신경망 내에서, 링크를 통해 연결된 하나 이상의 노드들은 상대적으로 입력 노드 및 출력 노드의 관계를 형성할 수 있다. 입력 노드 및 출력 노드의 개념은 상대적인 것으로서, 하나의 노드에 대하여 출력 노드 관계에 있는 임의의 노드는 다른 노드와의 관계에서 입력 노드 관계에 있을 수 있으며, 그 역도 성립할 수 있다. 상술한 바와 같이, 입력 노드 대 출력 노드 관계는 링크를 중심으로 생성될 수 있다. 하나의 입력 노드에 하나 이상의 출력 노드가 링크를 통해 연결될 수 있으며, 그 역도 성립할 수 있다.
- [0046] 하나의 링크를 통해 연결된 입력 노드 및 출력 노드 관계에서, 출력 노드의 데이터는 입력 노드에 입력된 데이터에 기초하여 그 값이 결정될 수 있다. 여기서 입력 노드와 출력 노드를 상호 연결하는 링크는 가중치(weight)를 가질 수 있다. 가중치는 가변적일 수 있으며, 신경망이 원하는 기능을 수행하기 위해, 사용자 또는 알고리즘에 의해 가변 될 수 있다. 예를 들어, 하나의 출력 노드에 하나 이상의 입력 노드가 각각의 링크에 의해 상호 연결된 경우, 출력 노드는 상기 출력 노드와 연결된 입력 노드들에 입력된 값들 및 각각의 입력 노드들에 대응하는 링크에 설정된 가중치에 기초하여 출력 노드 값을 결정할 수 있다.
- [0047] 상술한 바와 같이, 신경망은 하나 이상의 노드들이 하나 이상의 링크를 통해 상호 연결되어 신경망 내에서 입력

노드 및 출력 노드 관계를 형성한다. 신경망 내에서 노드들과 링크들의 개수 및 노드들과 링크들 사이의 연관관계, 링크들 각각에 부여된 가중치의 값에 따라, 신경망의 특성이 결정될 수 있다. 예를 들어, 동일한 개수의 노드 및 링크들이 존재하고, 링크들의 가중치 값이 상이한 두 신경망이 존재하는 경우, 두 개의 신경망들은 서로 상이한 것으로 인식될 수 있다.

[0048] 신경망은 하나 이상의 노드들의 집합으로 구성될 수 있다. 신경망을 구성하는 노드들의 부분 집합은 레이어(layer)를 구성할 수 있다. 신경망을 구성하는 노드들 중 일부는, 최초 입력 노드로부터의 거리들에 기초하여, 하나의 레이어(layer)를 구성할 수 있다. 예를 들어, 최초 입력 노드로부터 거리가 n 인 노드들의 집합은, n 레이어를 구성할 수 있다. 최초 입력 노드로부터 거리는, 최초 입력 노드로부터 해당 노드까지 도달하기 위해 거쳐야 하는 링크들의 최소 개수에 의해 정의될 수 있다. 그러나, 이러한 레이어의 정의는 설명을 위한 임의적인 것으로서, 신경망 내에서 레이어의 차수는 상술한 것과 상이한 방법으로 정의될 수 있다. 예를 들어, 노드들의 레이어는 최종 출력 노드로부터 거리에 의해 정의될 수도 있다.

[0049] 본 개시내용의 일 실시예에서, 뉴런들 또는 노드들의 집합은 레이어라는 표현으로 정의될 수 있다.

[0050] 최초 입력 노드는 신경망 내의 노드들 중 다른 노드들과의 관계에서 링크를 거치지 않고 데이터가 직접 입력되는 하나 이상의 노드들을 의미할 수 있다. 또는, 신경망 네트워크 내에서, 링크를 기준으로 한 노드 간의 관계에 있어서, 링크로 연결된 다른 입력 노드들을 가지지 않는 노드들을 의미할 수 있다. 이와 유사하게, 최종 출력 노드는 신경망 내의 노드들 중 다른 노드들과의 관계에서, 출력 노드를 가지지 않는 하나 이상의 노드들을 의미할 수 있다. 또한, 히든 노드는 최초 입력 노드 및 최종 출력 노드가 아닌 신경망을 구성하는 노드들을 의미할 수 있다.

[0051] 본 개시의 일 실시예에 따른 신경망은 입력 레이어의 노드의 개수가 출력레이어의 노드의 개수와 동일할 수 있으며, 입력 레이어에서 히든 레이어로 진행됨에 따라 노드의 수가 감소하다가 다시 증가하는 형태의 신경망일 수 있다. 또한, 본 개시의 다른 일 실시예에 따른 신경망은 입력 레이어의 노드의 개수가 출력 레이어의 노드의 개수 보다 적을 수 있으며, 입력 레이어에서 히든 레이어로 진행됨에 따라 노드의 수가 감소하는 형태의 신경망일 수 있다. 또한, 본 개시의 또 다른 일 실시예에 따른 신경망은 입력 레이어의 노드의 개수가 출력 레이어의 노드의 개수보다 많을 수 있으며, 입력 레이어에서 히든 레이어로 진행됨에 따라 노드의 수가 증가하는 형태의 신경망일 수 있다. 본 개시의 또 다른 일 실시예에 따른 신경망은 상술한 신경망들의 조합된 형태의 신경망일 수 있다.

[0052] 본 개시내용의 일 실시예에 따른 인공지능 기반 모델은 딥 뉴럴 네트워크(DNN: deep neural network, 심층신경망)를 포함할 수 있다. 딥 뉴럴 네트워크는 입력 레이어와 출력 레이어 외에 복수의 히든 레이어를 포함하는 신경망을 의미할 수 있다. 딥 뉴럴 네트워크를 이용하면 데이터의 잠재적인 구조(latent structures)를 파악할 수 있다. 즉, 사진, 글, 비디오, 음성, 단백질 시퀀스 구조, 유전자 시퀀스 구조, 펩타이드 서열의 구조, 음악의 잠재적인 구조(예를 들어, 어떤 물체가 사진에 있는지, 글의 내용과 감정이 무엇인지, 음성의 내용과 감정이 무엇인지 등), 및/또는 펩타이드와 MHC 간의 결합 친화도를 파악할 수 있다. 딥 뉴럴 네트워크는 컨볼루션 뉴럴 네트워크(CNN: convolutional neural network), 리커런트 뉴럴 네트워크(RNN: recurrent neural network), 오토인코더(AE: Autoencoder), 변이형 오토인코더(VAE: Variational Autoencoder), 제한 볼츠만 머신(RBM: restricted boltzmann machine), 심층 신뢰 네트워크(DBN: deep belief network), Q 네트워크, U 네트워크, 삼 네트워크, 적대적 생성 네트워크(GAN: Generative Adversarial Network), 트랜스포머 등을 포함할 수 있다. 전술한 딥 뉴럴 네트워크의 기재는 예시일 뿐이며 본 개시는 이에 제한되지 않는다.

[0053] 본 개시내용의 인공지능 기반 모델은 입력 레이어, 히든 레이어 및 출력 레이어를 포함하는 전술한 임의의 구조의 네트워크 구조에 의해 표현될 수 있다.

[0054] 본 개시내용의 인공지능 기반 모델에서 사용될 수 있는 뉴럴 네트워크는 지도 학습(supervised learning), 비지도 학습(unsupervised learning), 반지도학습(semi supervised learning), 전이 학습(transfer learning), 능동 학습(active learning) 또는 강화학습(reinforcement learning) 중 적어도 하나의 방식으로 학습될 수 있다. 뉴럴 네트워크의 학습은 뉴럴 네트워크가 특정한 동작을 수행하기 위한 지식을 뉴럴 네트워크에 적용하는 과정일 수 있다.

[0055] 뉴럴 네트워크는 출력의 오류를 최소화하는 방향으로 학습될 수 있다. 뉴럴 네트워크의 학습에서 반복적으로 학습 데이터를 뉴럴 네트워크에 입력시키고 학습 데이터에 대한 뉴럴 네트워크의 출력과 타겟의 에러를 계산하고, 에러를 줄이기 위한 방향으로 뉴럴 네트워크의 에러를 뉴럴 네트워크의 출력 레이어에서부터 입력 레이어 방향

으로 역전파(backpropagation)하여 뉴럴 네트워크의 각 노드의 가중치를 업데이트 하는 과정이다. 지도 학습의 경우 각각의 학습 데이터에 정답이 라벨링되어있는 학습 데이터를 사용하며(즉, 라벨링된 학습 데이터), 비지도 학습의 경우는 각각의 학습 데이터에 정답이 라벨링되어 있지 않을 수 있다. 즉, 예를 들어 데이터 분류에 관한 지도 학습의 경우의 학습 데이터는 학습 데이터 각각에 카테고리가 라벨링 된 데이터 일 수 있다. 라벨링된 학습 데이터가 뉴럴 네트워크에 입력되고, 뉴럴 네트워크의 출력(카테고리)과 학습 데이터의 라벨을 비교함으로써 오류(error)가 계산될 수 있다. 다른 예로, 데이터 분류에 관한 비지도 학습의 경우 입력인 학습 데이터가 뉴럴 네트워크 출력과 비교됨으로써 오류가 계산될 수 있다. 계산된 오류는 뉴럴 네트워크에서 역방향(즉, 출력 레이어에서 입력 레이어 방향)으로 역전파 되며, 역전파에 따라 뉴럴 네트워크의 각 레이어의 각 노드들의 연결 가중치가 업데이트 될 수 있다. 업데이트 되는 각 노드의 연결 가중치는 학습률(learning rate)에 따라 변화량이 결정될 수 있다. 입력 데이터에 대한 뉴럴 네트워크의 계산과 에러의 역전파는 학습 사이클(epoch)을 구성할 수 있다. 학습률은 뉴럴 네트워크의 학습 사이클의 반복 횟수에 따라 상이하게 적용될 수 있다. 예를 들어, 뉴럴 네트워크의 학습 초기에는 높은 학습률을 사용하여 뉴럴 네트워크가 빠르게 일정 수준의 성능을 확보하도록 하여 효율성을 높이고, 학습 후기에는 낮은 학습률을 사용하여 정확도를 높일 수 있다.

[0056] 뉴럴 네트워크의 학습에서 일반적으로 학습 데이터는 실제 데이터(즉, 학습된 뉴럴 네트워크를 이용하여 처리하고자 하는 데이터)의 부분집합일 수 있으며, 따라서, 학습 데이터에 대한 오류는 감소하나 실제 데이터에 대해서는 오류가 증가하는 학습 사이클이 존재할 수 있다. 과적합(overfitting)은 이와 같이 학습 데이터에 과하게 학습하여 실제 데이터에 대한 오류가 증가하는 현상이다. 예를 들어, 노란색 고양이를 보여 고양이를 학습한 뉴럴 네트워크가 노란색 이외의 고양이를 보고는 고양이임을 인식하지 못하는 현상이 과적합의 일종일 수 있다. 과적합은 머신러닝 알고리즘의 오류를 증가시키는 원인으로 작용할 수 있다. 이러한 과적합을 막기 위하여 다양한 최적화 방법이 사용될 수 있다. 과적합을 막기 위해서는 학습 데이터를 증가시키거나, 레귤라리제이션(regularization), 학습의 과정에서 네트워크의 노드 일부를 비활성화하는 드롭아웃(dropout), 배치 정규화 레이어(batch normalization layer)의 활용 등의 방법이 적용될 수 있다.

[0057] 본 개시의 일 실시예에 따른 데이터 구조를 저장한 컴퓨터 판독가능 매체가 개시된다. 전술한 데이터 구조는 본 개시내용에서의 저장부에 저장될 수 있으며, 프로세서에 의해 실행될 수 있으며 그리고 통신부에 의해 송수신될 수 있다.

[0058] 데이터 구조는 데이터에 효율적인 접근 및 수정을 가능하게 하는 데이터의 조직, 관리, 저장을 의미할 수 있다. 데이터 구조는 특정 문제(예를 들어, 데이터 분석, 데이터 검색, 데이터 저장, 데이터 수정) 해결을 위한 데이터의 조직을 의미할 수 있다. 데이터 구조는 특정한 데이터 처리 기능을 지원하도록 설계된, 데이터 요소들 간의 물리적이거나 논리적인 관계로 정의될 수도 있다. 데이터 요소들 간의 논리적인 관계는 사용자 정의 데이터 요소들 간의 연결관계를 포함할 수 있다. 데이터 요소들 간의 물리적인 관계는 컴퓨터 판독가능 저장매체(예를 들어, 영구 저장 장치)에 물리적으로 저장되어 있는 데이터 요소들 간의 실제 관계를 포함할 수 있다. 데이터 구조는 구체적으로 데이터의 집합, 데이터 간의 관계, 데이터에 적용할 수 있는 함수 또는 명령어를 포함할 수 있다. 효과적으로 설계된 데이터 구조를 통해 컴퓨팅 장치는 컴퓨팅 장치의 자원을 최소한으로 사용하면서 연산을 수행할 수 있다. 구체적으로 컴퓨팅 장치는 효과적으로 설계된 데이터 구조를 통해 연산, 읽기, 삽입, 삭제, 비교, 교환 및 검색의 효율성을 높일 수 있다.

[0059] 데이터 구조는 데이터 구조의 형태에 따라 선형 데이터 구조와 비선형 데이터 구조로 구분될 수 있다. 선형 데이터 구조는 하나의 데이터 뒤에 하나의 데이터만이 연결되는 구조일 수 있다. 선형 데이터 구조는 리스트(List), 스택(Stack), 큐(Queue), 데크(Deque)를 포함할 수 있다. 리스트는 내부적으로 순서가 존재하는 일련의 데이터 집합을 의미할 수 있다. 리스트는 연결 리스트(Linked List)를 포함할 수 있다. 연결 리스트는 각각의 데이터가 포인터를 가지고 한 줄로 연결되어 있는 방식으로 데이터가 연결된 데이터 구조일 수 있다. 연결 리스트에서 포인터는 다음이나 이전 데이터와의 연결 정보를 포함할 수 있다. 연결 리스트는 형태에 따라 단일 연결 리스트, 이중 연결 리스트, 원형 연결 리스트로 표현될 수 있다. 스택은 제한적으로 데이터에 접근할 수 있는 데이터 나열 구조일 수 있다. 스택은 데이터 구조의 한 쪽 끝에서만 데이터를 처리(예를 들어, 삽입 또는 삭제)할 수 있는 선형 데이터 구조일 수 있다. 스택에 저장된 데이터는 늦게 들어갈수록 빨리 나오는 데이터 구조(LIFO-Last in First Out)일 수 있다. 큐는 제한적으로 데이터에 접근할 수 있는 데이터 나열 구조로서, 스택과 달리 늦게 저장된 데이터일수록 늦게 나오는 데이터 구조(FIFO-First in First Out)일 수 있다. 데크는 데이터 구조의 양 쪽 끝에서 데이터를 처리할 수 있는 데이터 구조일 수 있다.

[0060] 비선형 데이터 구조는 하나의 데이터 뒤에 복수개의 데이터가 연결되는 구조일 수 있다. 비선형 데이터 구조는 그래프(Graph) 데이터 구조를 포함할 수 있다. 그래프 데이터 구조는 정점(Vertex)과 간선(Edge)으로 정의될 수

있으며 간선은 서로 다른 두개의 정점을 연결하는 선을 포함할 수 있다. 그래프 데이터 구조 트리(Tree) 데이터 구조를 포함할 수 있다. 트리 데이터 구조는 트리에 포함된 복수개의 정점 중에서 서로 다른 두개의 정점을 연결시키는 경로가 하나인 데이터 구조일 수 있다. 즉 그래프 데이터 구조에서 루프(loop)를 형성하지 않는 데이터 구조일 수 있다.

[0061] 본 명세서에 걸쳐, 인공지능 기반 모델, 연산 모델, 신경망, 네트워크 함수, 뉴럴 네트워크(neural network)는 상호 교환 가능한 의미로 사용될 수 있다. 이하에서는 신경망으로 통일하여 기술한다. 데이터 구조는 신경망을 포함할 수 있다. 그리고 신경망을 포함한 데이터 구조는 컴퓨터 판독가능 매체에 저장될 수 있다. 신경망을 포함한 데이터 구조는 또한 신경망에 의한 처리를 위하여 전처리된 데이터, 신경망에 입력되는 데이터, 신경망의 가중치, 신경망의 하이퍼 파라미터, 신경망으로부터 획득한 데이터, 신경망의 각 노드 또는 레이어와 연관된 활성 함수, 신경망의 학습을 위한 손실 함수 등을 포함할 수 있다. 신경망을 포함한 데이터 구조는 상기 개시된 구성들 중 임의의 구성 요소들을 포함할 수 있다. 즉 신경망을 포함한 데이터 구조는 신경망에 의한 처리를 위하여 전처리된 데이터, 신경망에 입력되는 데이터, 신경망의 가중치, 신경망의 하이퍼 파라미터, 신경망으로부터 획득한 데이터, 신경망의 각 노드 또는 레이어와 연관된 활성 함수, 신경망의 학습을 위한 손실 함수 등 전부 또는 이들의 임의의 조합을 포함하여 구성될 수 있다. 전술한 구성들 이외에도, 신경망을 포함한 데이터 구조는 신경망의 특성을 결정하는 임의의 다른 정보를 포함할 수 있다. 또한, 데이터 구조는 신경망의 연산 과정에 사용되거나 발생하는 모든 형태의 데이터를 포함할 수 있으며 전술한 사항에 제한되는 것은 아니다. 컴퓨터 판독가능 매체는 컴퓨터 판독가능 기록 매체 및/또는 컴퓨터 판독가능 전송 매체를 포함할 수 있다. 신경망은 일반적으로 노드라 지칭될 수 있는 상호 연결된 계산 단위들의 집합으로 구성될 수 있다. 이러한 노드들은 뉴런(neuron)들로 지칭될 수도 있다. 신경망은 적어도 하나 이상의 노드들을 포함하여 구성된다.

[0062] 데이터 구조는 신경망에 입력되는 데이터를 포함할 수 있다. 신경망에 입력되는 데이터를 포함하는 데이터 구조는 컴퓨터 판독가능 매체에 저장될 수 있다. 신경망에 입력되는 데이터는 신경망 학습 과정에서 입력되는 학습 데이터 및/또는 학습이 완료된 신경망에 입력되는 입력 데이터를 포함할 수 있다. 신경망에 입력되는 데이터는 전처리(pre-processing)를 거친 데이터 및/또는 전처리 대상이 되는 데이터를 포함할 수 있다. 전처리는 데이터를 신경망에 입력시키기 위한 데이터 처리 과정을 포함할 수 있다. 따라서 데이터 구조는 전처리 대상이 되는 데이터 및 전처리로 발생하는 데이터를 포함할 수 있다. 전술한 데이터 구조는 예시일 뿐 본 개시는 이에 제한되지 않는다.

[0063] 데이터 구조는 신경망의 가중치를 포함할 수 있다. (본 명세서에서 가중치, 파라미터는 상호 교환가능한 의미로 사용될 수 있다.) 그리고 신경망의 가중치를 포함한 데이터 구조는 컴퓨터 판독가능 매체에 저장될 수 있다. 신경망은 복수개의 가중치를 포함할 수 있다. 가중치는 가변적일 수 있으며, 신경망이 원하는 기능을 수행하기 위해, 사용자 또는 알고리즘에 의해 가변 될 수 있다. 예를 들어, 하나의 출력 노드에 하나 이상의 입력 노드가 각각의 링크에 의해 상호 연결된 경우, 출력 노드는 상기 출력 노드와 연결된 입력 노드들에 입력된 값들 및 각각의 입력 노드들에 대응하는 링크에 설정된 가중치에 기초하여 출력 노드에서 출력되는 데이터 값을 결정할 수 있다. 전술한 데이터 구조는 예시일 뿐 본 개시는 이에 제한되지 않는다.

[0064] 제한이 아닌 예로서, 가중치는 신경망 학습 과정에서 가변되는 가중치 및/또는 신경망 학습이 완료된 가중치를 포함할 수 있다. 신경망 학습 과정에서 가변되는 가중치는 학습 사이클이 시작되는 시점의 가중치 및/또는 학습 사이클 동안 가변되는 가중치를 포함할 수 있다. 신경망 학습이 완료된 가중치는 학습 사이클이 완료된 가중치를 포함할 수 있다. 따라서 신경망의 가중치를 포함한 데이터 구조는 신경망 학습 과정에서 가변되는 가중치 및/또는 신경망 학습이 완료된 가중치를 포함한 데이터 구조를 포함할 수 있다. 그러므로 상술한 가중치 및/또는 각 가중치의 조합은 신경망의 가중치를 포함한 데이터 구조에 포함되는 것으로 한다. 전술한 데이터 구조는 예시일 뿐 본 개시는 이에 제한되지 않는다.

[0065] 신경망의 가중치를 포함한 데이터 구조는 직렬화(serialization) 과정을 거친 후 컴퓨터 판독가능 저장 매체(예를 들어, 메모리, 하드 디스크)에 저장될 수 있다. 직렬화는 데이터 구조를 동일하거나 다른 컴퓨팅 장치에 저장하고 나중에 다시 재구성하여 사용할 수 있는 형태로 변환하는 과정일 수 있다. 컴퓨팅 장치는 데이터 구조를 직렬화하여 네트워크를 통해 데이터를 송수신할 수 있다. 직렬화된 신경망의 가중치를 포함한 데이터 구조는 역직렬화(deserialization)를 통해 동일한 컴퓨팅 장치 또는 다른 컴퓨팅 장치에서 재구성될 수 있다. 신경망의 가중치를 포함한 데이터 구조는 직렬화에 한정되는 것은 아니다. 나아가 신경망의 가중치를 포함한 데이터 구조는 컴퓨팅 장치의 자원을 최소한으로 사용하면서 연산의 효율을 높이기 위한 데이터 구조(예를 들어, 비선형 데이터 구조에서 B-Tree, R-Tree, Trie, m-way search tree, AVL tree, Red-Black Tree)를 포함할 수 있다. 전술

한 사항은 예시일 뿐 본 개시는 이에 제한되지 않는다.

- [0066] 데이터 구조는 신경망의 하이퍼 파라미터(Hyper-parameter)를 포함할 수 있다. 그리고 신경망의 하이퍼 파라미터를 포함한 데이터 구조는 컴퓨터 판독가능 매체에 저장될 수 있다. 하이퍼 파라미터는 사용자에 의해 가변되는 변수일 수 있다. 하이퍼 파라미터는 예를 들어, 학습률(learning rate), 비용 함수(cost function), 학습 사이클 반복 횟수, 가중치 초기화(Weight initialization)(예를 들어, 가중치 초기화 대상이 되는 가중치 값의 범위 설정), Hidden Unit 개수(예를 들어, 히든 레이어의 개수, 히든 레이어의 노드 수)를 포함할 수 있다. 전술한 데이터 구조는 예시일 뿐 본 개시는 이에 제한되지 않는다.
- [0067] 본 개시의 일 실시예에 따른 인공지능 모델은 생성모델(Generative model)일 수 있다. 생성모델은 주어진 학습 데이터의 분포를 학습하여, 학습 데이터의 분포를 따르는 유사한 데이터를 생성하는 모델이다. 생성모델은 변이형 오토인코더(VAE), 적대적 생성 네트워크(GAN), 디퓨전 모델(Diffusion model) 등을 포함할 수 있다. 변이형 오토인코더(VAE)는 학습 데이터의 분포를 추정하고(approximate density), 추정된 학습 데이터의 분포를 기반으로 입력 데이터를 재구성하도록 학습하는 모델을 의미할 수 있다. 적대적 생성 네트워크(GAN)는 학습 데이터의 분포를 명시적으로 정의하지 않고도 데이터를 생성하도록 학습하는 모델을 의미할 수 있다. 디퓨전 모델은 고정된 정규 분포로부터 생성된 노이즈를 데이터에 점진적으로 추가하는 확산 프로세스(diffusion process)를 포함할 수 있다. 또한, 디퓨전 모델은 노이즈를 제거하는 역확산 프로세스(reverse process)를 학습함으로써 데이터를 재구성할 수 있다. 즉, 디퓨전 모델은 학습된 정규 분포로부터 생성된 노이즈를 점진적으로 제거함으로써 입력 데이터와 유사한 확률 분포를 가진 결과 이미지를 생성하는 모델을 의미할 수 있다.
- [0068] 일반적으로 오토인코더는 레이블링(labeling)되지 않은 데이터로부터 저차원의 특징(low-dimensional feature representation)을 학습하는 비지도 학습을 수행할 수 있다. 여기서 인코더가 추출한 특징은 특정 숫자값으로 정의된 잠재 코드(latent code)이다.
- [0069] 본 개시의 일 실시예에 따른 모델들은 가우시안 분포 노이즈(Gaussian distributed noise)에 대하여 예측된 노이즈의 부분 또는 전부를 제거하여, 상기 예측된 노이즈의 부분 또는 전부가 제거된 데이터를 획득하도록 학습될 수 있다.
- [0070] 본 개시의 일 실시예에 따른 컴퓨팅 장치(100)는 등방성 가우시안 분포 노이즈(isotropic gaussian distributed noise)에 대하여 상기 노이즈의 부분 또는 전부를 제거하여, 상기 노이즈의 부분 또는 전부가 제거된 데이터를 획득하도록 뉴럴 네트워크를 학습시킬 수 있다. 이때, 상기 뉴럴 네트워크는 조건부(conditional) 노이즈 예측 모델을 포함할 수 있다. 또한, 조건부 노이즈 예측 모델은 입력과 출력이 동일한 크기를 가지는 U-Net 구조를 포함할 수 있고, 노이즈가 포함된 데이터 $x(t)$ 와 확산(diffusion) 타임 스텝(time step) t 를 입력으로 하고, 노이즈가 포함된 데이터 $x(t)$ 에 포함된 확산 노이즈(diffusion noise)를 예측하여 출력할 수 있다.
- [0071] 컴퓨팅 장치(100)는 노이즈가 포함되지 않은 원본 데이터 $x(0)$ 에 대하여 T번의 타임 스텝(time step)에 걸쳐 조금씩 랜덤 가우시안 노이즈를 추가하는 과정을 반복하고, 결과적으로 등방성 가우시안 분포 노이즈 $x(T)$ (isotropic gaussian distributed noise)를 획득하는 정방향 과정을 수행할 수 있다. 상기 원본 데이터 $x(0)$ 의 종류는 이미지 데이터 이외에도 오디오 데이터 등 다양한 예시들이 포함될 수 있다. 한편, 본 개시의 일 실시예에 따른 정방향 과정은 예를 들어 다음과 같은 수식을 통해 수행될 수 있다.

수학식 1

[0072]
$$x_t = \sqrt{1 - \beta_t} x_{t-1} + \sqrt{\beta_t} \varepsilon$$

[0073]
$$\varepsilon \sim N(0, I)$$

- [0074] 상기 수학식 1에서 β_t 는 하이퍼 파라미터로써 확산 계수를 계산하는 과정에서 이용될 수 있고, β_t 는 임의의 값으로 설정될 수 있으며 $0 < \beta_1 < \beta_2 < \dots < \beta_T < 1$ 의 값으로 설정될 수 있다. 예를 들어 β_1 이 0.0001의 값을 가지고 β_T 가 0.02의 값을 가질 수 있고, β_1 부터 β_T 까지 β_t 의 값은 선형적으로(linear) 증가할 수 있고, 코사인 함수를 따라 증가할 수 있으며, 총 확산 스텝 수를 의미하는 T는 1000으로 설정될 수 있다. 다만, β_t 가 선형적으로 또는 코사인 함수를 따라 증가되는 것은 예시에 불과할 뿐, 본 개시의 실시예들에 따르면 상기 원본 데이터의

종류에 따라 β_ε 의 증가량이 다르게 결정될 수 있으며, 구체적인 설명은 이하 [수학식 2]를 통해 후술된다. 또한, ε 은 랜덤 가우시안 분포 노이즈를 의미할 수 있다. 타임 스텝 t에서 상기 노이즈가 포함된 데이터 x(t)을 노이즈가 포함되지 않은 원본 데이터 x(0)와 포함된 확산 노이즈(diffusion noise)에 관한 식으로 나타낸 일반식은 다음과 같이 표현될 수 있다.

수학식 2

$$\alpha_\varepsilon = 1 - \beta_\varepsilon \quad \dots (1)$$

$$\bar{\alpha}_t = \prod_{i=1}^t \alpha_i \quad \dots (2)$$

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon \quad \dots (3)$$

$$n_t = \sqrt{1 - \bar{\alpha}_t}, s_t = \sqrt{\bar{\alpha}_t} \quad \dots (4)$$

본 개시의 일 실시예에 따라 상기 수학식 2에서 (1)번 식과 (2)번 식은 확산 계수 $\bar{\alpha}_\varepsilon$ 의 구체적인 의미를 표현하는 식이다. 상기 수학식 2의 (1)번 식에서 특정 타임 스텝 t에서 확산 계수 α_ε 는 1에서 상기 하이퍼 파라미터 β_ε 를 뺀 값으로 계산될 수 있으며, 상기 수학식 2의 (2)번 식에서 확산 계수 $\bar{\alpha}_\varepsilon$ 는 타임 스텝 1부터 t까지 순차적으로 누적된 확산 계수를 의미할 수 있다. 따라서, 상기 수학식 2에서 (3)번 식은 타임 스텝 t에서 상기 노이즈가 포함된 데이터 x(t)를 노이즈가 포함되지 않은 원본 데이터 x(0)와 확산 계수($\bar{\alpha}_\varepsilon$)와 랜덤 가우시안 분포 노이즈(ε)에 관한 수식으로 나타낸 것이다. 또한, 상기 수학식 2에서 (4)번 식은 본 개시의 일 실시예에 따라 상기 원본 데이터의 종류에 기초하여 결정되는 노이즈의 비율n(t) 및 시그널의 비율s(t)을 의미할 수 있다. 따라서, 타임 스텝 t에서 노이즈가 포함된 데이터 x(t)는 노이즈의 비율n(t) 및 시그널의 비율s(t)에 기초하여 표현될 수 있다. 구체적으로, 타임 스텝 t에서 노이즈가 포함된 데이터 x(t)는 노이즈가 포함되지 않은 원본 데이터 x(0)의 비율(즉, 시그널의 비율s(t))이 높아질수록 원본 데이터의 형태에 가까워지고, 랜덤 가우시안 분포 노이즈(ε)의 비율(즉, 노이즈의 비율n(t))이 높아질수록 랜덤 가우시안 분포 노이즈의 형태에 가까워질 수 있다.

예를 들어, 컴퓨팅 장치(100)는 노이즈가 포함되지 않은 원본 데이터 x(0)에 대하여 T번의 타임 스텝(time step)에 걸쳐 조금씩 랜덤 가우시안 노이즈를 추가하는 정방향 과정에서 상기 결정된 노이즈의 비율에 기초하여 상기 원본 데이터에 제1 노이즈를 추가하여 제1 데이터를 획득하고, 상기 결정된 노이즈의 비율에 기초하여 상기 제1 데이터에 제2 노이즈를 추가하여 제2 데이터를 획득할 수 있다. 일 실시예에서, 상기 원본 데이터 x(0)의 종류는 이미지 데이터 이외에도 텍스트 데이터 및 오디오 데이터 등 다양한 예시들이 포함될 수 있다.

이를 통해, 컴퓨팅 장치(100)는 노이즈가 포함되지 않은 원본 데이터 x(0)에 대하여 “상기 수학식 2에서 (4)번 식을 통해 결정되는 노이즈의 비율”에 기초하여 T번의 타임 스텝(time step)에 걸쳐 랜덤 가우시안 노이즈를 추가하는 과정을 반복하고, 결과적으로 등방성 가우시안 분포 노이즈x(T)(isotropic gaussian distributed noise)를 획득하는 정방향 과정을 수행할 수 있다. 다만, 상기 정방향 과정은 상기 수학식 2의 예시에 한정되지 않고, 데이터에 노이즈를 추가하는 다양한 과정들이 정방향 과정에 포함될 수 있다.

또한, 컴퓨팅 장치(100)는 상기 정방향 과정과 반대 방향으로 등방성 가우시안 분포 노이즈x(T)(isotropic gaussian distributed noise)에 대하여 T번의 타임 스텝(time step)에 걸쳐 조금씩 랜덤 가우시안 노이즈를 제거하는 과정을 반복하고, 결과적으로 노이즈가 포함되지 않은 원본 데이터 x(0)를 획득하는 역방향 과정을 수행하도록 상기 뉴럴 네트워크를 학습시킬 수 있다. 이와 관련하여, 상기 역방향 수행 과정을 나타낸 수식은 다음과 같이 표현될 수 있다.

수학식 3

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \varepsilon_{\theta}(x_t, t) \right) + \sigma_t \varepsilon \quad \dots (1)$$

$$\alpha_t = \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}} \quad \dots (2)$$

$$\sigma_t = 1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}} \quad \dots (3)$$

상기 수학식 3에서 (1)번 식은 “노이즈가 포함된 데이터 $x(t)$ ”에 대하여 상기 뉴럴 네트워크가 예측한 노이즈 예측 결과($\varepsilon_{\theta}(x_t, t)$)를 제거하여 “상기 노이즈가 일부 제거된 전 단계의 데이터 $x(t-1)$ ”를 획득하는 역방향 과정을 나타내는 수식이다. 예를 들어, 컴퓨팅 장치(100)는 “제2 노이즈가 포함된 제2 데이터 $x(2)$ ”에 대하여 상기 뉴럴 네트워크가 예측한 노이즈 예측 결과를 제거하여 “상기 제2 노이즈가 제거된 전 단계의 제1 데이터 $x(1)$ ”를 획득할 수 있다. 상기 수학식 3에서 (2)번 식은 현재 타임 스텝 t 에서의 확산 계수 α_t 를 의미하고, 상기 수학식 3에서 (3)번 식에서 σ_t 는 분산 파라미터를 의미하고, 확산 계수 α_t 에 기초하여 계산될 수 있다. 다만, 상기 역방향 과정은 상기 [수학식 3]에 한정되지 않고, 노이즈가 포함된 데이터에 대해 노이즈를 제거하는 다양한 과정들이 역방향 과정에 포함될 수 있다.

구체적으로, 컴퓨팅 장치(100)는 뉴럴 네트워크에 노이즈가 포함된 데이터 $x(t)$ 와 타임 스텝 t 를 입력하고, 상기 뉴럴 네트워크가 예측한 노이즈 예측 결과와 실제로 포함된 확산 노이즈(diffusion noise)를 비교하여 손실 함수를 계산하고, 상기 손실 함수에 따라 경사 하강(gradient descent)을 수행하여 상기 뉴럴 네트워크를 학습시킬 수 있다. 예를 들어, 상기 뉴럴 네트워크가 예측한 노이즈 예측 결과와 실제로 포함된 확산 노이즈(diffusion noise)를 비교하여 계산되는 손실 함수는 다음과 같은 수식으로 표현될 수 있다.

수학식 4

$$\text{노이즈 예측 손실} = E_{\varepsilon, x_0, \varepsilon} [\|\varepsilon - \varepsilon_{\theta}(x_t, t)\|^2]$$

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon$$

상기 [수학식 4]에서 손실 함수는 실제로 포함된 확산 노이즈(diffusion noise) (ε)과 상기 뉴럴 네트워크가 예측한 노이즈 예측 결과($\varepsilon_{\theta}(x_t, t)$)를 비교하여 계산될 수 있다. 예를 들어, 컴퓨팅 장치(100)는 “상기 제1 데이터 $x(1)$ 에 포함된 제1 노이즈”를 예측하고, 상기 예측된 제1 노이즈 및 상기 원본 데이터 $x(0)$ 에 추가된 상기 제1 노이즈를 비교하여 제1 손실 함수를 계산할 수 있다. 컴퓨팅 장치(100)는 “상기 제2 데이터 $x(2)$ 에 포함된 제2 노이즈”를 예측하고, 상기 예측된 제2 노이즈 및 상기 제1 데이터 $x(1)$ 에 추가된 상기 제2 노이즈를 비교하여 제2 손실 함수를 계산할 수 있다. 다만, 상기 손실 함수는 수학식 4의 예시에 한정되지 않고, 노이즈 예측 결과와 실제로 포함된 확산 노이즈를 비교하여 계산된 다양한 손실 함수들이 포함될 수 있다.

추가적으로, 상기 뉴럴 네트워크는 상기 제2 노이즈가 포함된 데이터 $x(2)$ 에 포함된 확산 노이즈를 예측하고, 상기 예측된 제2 노이즈를 제거함으로써 “제2 노이즈가 제거된 데이터 $x(1)$ ”를 획득하도록 학습될 수 있다. 예를 들어, 또한, 컴퓨팅 장치(100)는 상기 제1 손실 함수 또는 상기 제2 손실 함수 중 적어도 하나에 기초하여 상기 뉴럴 네트워크를 학습시킬 수 있다. 또한, 상기 뉴럴 네트워크는 “상기 노이즈가 포함된 데이터 $x(t)$ ”에 포함된 확산 노이즈를 예측하고, 상기 예측된 확산 노이즈를 제거하는 과정을 한번 이상 반복하여, “상기 노이즈가 포함된 데이터 $x(t)$ ”에 포함된 확산 노이즈를 전부 제거함으로써, 노이즈가 전부 제거된 원본 데이터 $x(0)$ 가 획득되도록 학습될 수 있다. 한편, 상기 뉴럴 네트워크를 활용하여 노이즈가 포함된 데이터에 대하여 확산 노이즈를 제거하도록 학습하는 과정에서 원본 데이터의 종류에 기초하여 노이즈의 비율이 결정됨으로써, 더 좋은 품질의 데이터를 생성하도록 상기 뉴럴 네트워크가 학습될 수 있다.

[0092] 컴퓨팅 장치(100)는 상기 결정된 원본 데이터의 종류에 대해 표현 가능한 데이터의 크기에 기초하여 노이즈의 비율을 결정할 수 있다. 구체적으로, 컴퓨팅 장치(100)는 상기 결정된 원본 데이터의 종류에 대해 표현 가능한 데이터의 최소 크기보다 상기 원본 데이터에 추가되는 노이즈의 크기가 작도록 상기 노이즈의 비율을 결정할 수 있다. 예를 들어, 상기 원본 데이터의 종류가 이미지(image) 데이터로 결정된 경우, 상기 원본 데이터는 채널 당 8비트(bit) 깊이(depth)를 가질 수 있다. 따라서, 컴퓨팅 장치(100)는 이미지 데이터에 대해 표현 가능한 데이터의 최소 크기인 $1/(2^8)$ (=약 0.0039)보다 상기 원본 데이터에 추가되는 노이즈의 크기가 작도록 상기 노이즈의 제1 비율을 결정할 수 있다.

[0093] 한편, 본 개시의 다른 실시예에 따르면 상기 원본 데이터의 종류가 오디오(audio) 데이터로 결정된 경우, 상기 원본 데이터는 하나의 샘플 당 16비트(bit) 깊이(depth)를 가질 수 있다. 따라서, 컴퓨팅 장치(100)는 오디오 데이터에 대해 표현 가능한 데이터의 최소 크기인 $1/(2^{16})$ (=약 0.000015)보다 상기 원본 데이터에 추가되는 노이즈의 크기가 작도록 상기 노이즈의 제2 비율을 결정할 수 있다. 따라서, 상기 원본 데이터가 이미지 데이터인 경우의 노이즈의 제1 비율보다 상기 원본 데이터가 오디오 데이터인 경우의 노이즈의 제2 비율이 더 작게 결정될 수 있다. 예를 들어, 상기 노이즈의 비율은 다음과 같은 수학적식을 통해 결정될 수 있다.

수학적식 5

[0094]
$$n_e = 2^{r^y} - 1 \dots (1)$$

[0095]
$$s_e = -(2^{r^y} - 1) + 1 \dots (2)$$

[0096] 구체적으로, 본 개시의 일 실시예에 따라 상기 [수학적식 5]에서 (1)번 식을 참조하면 노이즈의 비율 $n(t)$ 는 지수적으로 증가하도록 결정될 수 있고, 상기 [수학적식 5]에서 (2)번 식을 참조하면 시그널의 비율 $s(t)$ 는 지수적으로 감소하도록 결정될 수 있다. 이때, r 은 상기 노이즈 비율 $n(t)$ 및 상기 시그널 비율 $s(t)$ 의 곡선의 형태(shape)를 결정하는 하이퍼 파라미터를 의미할 수 있다. 다만, 상기 [수학적식 5]는 예시에 불과할 뿐 이외에도 노이즈의 비율 및 시그널의 비율은 다양하게 결정될 수 있다. 이를 통해, 이미지 데이터와 비교하여 상대적으로 노이즈에 민감한 데이터 도메인(예시적으로, 오디오 데이터 등)에서도 상기 신경망 모델을 통해 생성된 결과물이 노이즈의 영향이 거의 없도록 상기 신경망 모델이 학습될 수 있다.

[0097] 이하에서, 컴퓨팅 장치에서 수행되는, 이상 탐지를 위한 모델을 학습시키기 위한 방법을 후술하도록 한다.

[0098] 도 3은 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 이상 탐지를 위한 모델을 학습시키기 위한 방법을 나타낸 도면이다. 도 4는 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 이상 탐지를 위한 모델의 학습에 사용되는 데이터의 전처리 방법을 나타낸 도면이다. 일 실시예에서, 도 3 및/또는 도 4에서 도시되는 단계들 중 일부는 컴퓨팅 장치(100)에서 수행되고 다른 일부는 다른 컴퓨팅 장치에서 수행되는 방식과 같이 복수의 엔티티(entity)들에 의해 구현될 수도 있다. 도 3 및/또는 도 4에서 도시되는 단계들은 본 개시의 일 실시예에 따른 예시적인 단계로, 일부 단계를 추가되거나 생략할 수 있다. 도 3 및/또는 도 4에서 도시되는 단계들의 순서는 상황에 따라 달라질 수 있다.

[0099] 도 3을 참조하면, 컴퓨팅 장치(100)의 프로세서(110)는 무인 이동체와 관련된 복수의 센서들 각각에서 복수의 센서 데이터들을 획득할 수 있다(310).

[0100] 일 실시예에서, 무인 이동체는 사람이 직접 탑승하지 않고 원격 조종에 의해 제어되거나 또는 기기 외부의 환경을 스스로 인식하여 상황을 판단하고 임무를 수행할 수 있는 이동체일 수 있다. 예를 들어, 무인 이동체는 드론(drone), 자율주행 자동차 등일 수 있다. 일 실시예에서, 컴퓨팅 장치(100)는 무인 이동체에 대응되거나 또는 무인 이동체 내에 구비될 수 있다.

[0101] 일 실시예에서, 무인 이동체와 관련된 복수의 센서들은 무인 이동체 내의 정보, 무인 이동체를 둘러싼 주변 환경 정보 등을 센싱하기 위한 센서일 수 있다. 예를 들어, 무인 이동체와 관련된 복수의 센서들은 근접 센서(proximity sensor), 조도 센서(illumination sensor), 터치 센서(touch sensor), 가속도 센서(acceleration sensor), 자기 센서(magnetic sensor), 중력 센서(G-sensor), 자이로스코프 센서(gyroscope sensor), 모션 센서(motion sensor), 색상 감지 센서, 적외선 센서(IR 센서: infrared sensor), 초음파 센서(ultrasonic sensor), 광 센서(optical sensor, 예를 들어, 카메라 등), 마이크론(microphone), 배터리 게이지(battery gauge), 환경 센서(예를 들어, 기압계, 습도계, 온도계, 방사능 감지 센서, 열 감지 센서, 가스 감지 센서 등)

및/또는 화학 센서(예를 들어, 전자 코, 헬스케어 센서, 생체 인식 센서 등) 중 적어도 하나를 포함할 수 있다. 한편, 일 실시예에서, 컴퓨팅 장치(100)는 이러한 센서들 중 적어도 둘 이상의 센서에서 센싱되는 정보들을 조합하여 활용할 수 있다. 다만, 센서의 종류는 이에 한정되지 않는다.

- [0102] 일 실시예에서, 복수의 센서 데이터들은 상술한 복수의 센서들 각각에서 획득되는 데이터일 수 있다. 예를 들어, 복수의 센서 데이터들은 GPS(Global Positioning System) 정보, GNSS(Global Navigation Satellite System) 정보, 고도 정보, 가속도 정보, 배터리 정보, 수평 정보, 기압 정보, 온도 정보, 압력 정보, 습도 정보를 포함할 수 있다. 다만, 센서 데이터의 종류는 이에 한정되지 않는다.
- [0103] 일 실시예에서, 컴퓨팅 장치(100) 및/또는 무인 이동체는 하나의 센서로부터 적어도 하나의 센서데이터를 획득할 수 있다. 예를 들어, 컴퓨팅 장치(100) 및/또는 무인 이동체는 하나의 센서로부터 사전 결정된 시점(예를 들어, 1분 등)마다 센서 데이터를 획득할 수 있다. 따라서, 컴퓨팅 장치(100) 및/또는 무인 이동체는 센서마다 복수의 센서데이터들을 획득할 수도 있다.
- [0104] 일 실시예에서, 컴퓨팅 장치(100)는 무인 이동체로부터 복수의 센서데이터들을 수신할 수 있다.
- [0105] 프로세서(110)는 복수의 센서 데이터들을 시간 기준으로 정렬 및 압축하는 전처리를 수행할 수 있다(320).
- [0106] 도 4를 참조하면, 프로세서(110)는 복수의 센서 데이터들을 사전 결정된 시간 구간에 따라 분류할 수 있다(410). 예를 들어, 프로세서(110)는 복수의 센서 데이터들을 1~10분 구간, 11~20분 구간 등으로 분류할 수 있다. 따라서, 각 시간 구간별로 하나의 세트의 센서 데이터들이 존재할 수 있다. 예를 들어, 제1 시간 구간에는 제1 세트의 센서 데이터들이 존재할 수 있다. 제1 세트의 센서 데이터들은 제1 시간 구간에 제1 센서로부터 획득된 적어도 하나의 제1 센서 데이터, 제1 시간 구간에 제2 센서로부터 획득된 적어도 하나의 제2 센서 데이터, 제1 시간 구간에 제3 센서로부터 획득된 적어도 하나의 제3 센서데이터 등이 포함될 수 있다.
- [0107] 프로세서(110)는 사전 결정된 시간 구간별로 대응되는 센서 데이터의 변화량을 산출할 수 있다(420).
- [0108] 일 실시예에서, 프로세서(110)는 제1 시간 구간의 제1 세트의 센서 데이터들 중에서, 수치형(numerical) 센서 데이터를 생성하는 제1 센서로부터 획득된 제1 센서 데이터의 제1 변화량을 산출할 수 있다.
- [0109] 일 실시예에서, 수치형 센서 데이터는 특정 물리적, 화학적, 또는 환경적 특성을 수치로 측정한 데이터를 의미할 수 있다. 수치형 센서 데이터는 정량적으로 표현될 수 있다. 수치형 센서 데이터를 생성하는 제1 센서는 예를 들어, 온도 센서, 압력 센서 등을 포함할 수 있다.
- [0110] 본 개시에서 변화량(예를 들어, 제1 변화량 등)은 물리적, 화학적 또는 수학적 양이 시간이나 다른 변수에 따라 달라지는 정도를 나타내는 측정치일 수 있다. 일 실시예에서, 프로세서(110)는 모든 센서 값의 변화량을 계산하고 새로운 변수로 추가할 수 있다.
- [0111] 일 실시예에서, 제1 변화량은 제1 시간 구간보다 이전의 시간 구간에서 제1 센서로부터 획득된 이전 센서 데이터와, 제1 센서 데이터 간의 차이 값을 포함할 수 있다. 몇몇 실시예에서, 제1 변화량은 제1 시간 구간보다 이후의 시간 구간에서 제1 센서로부터 획득된 이후 센서 데이터와, 제1 센서 데이터 간의 차이 값을 포함할 수 있다.
- [0112] 일 실시예에서, 제1 변화량은 제1 센서 데이터가 복수인 경우, 복수의 제1 센서 데이터들 간의 차이 값을 포함할 수 있다. 예를 들어, 제1 변화량은 복수의 제1 센서 데이터들 중에서 제1-1 센서 데이터와 제1-2 센서 데이터 간의 차이 값을 포함할 수 있다.
- [0113] 프로세서(110)는 사전 결정된 시간 구간별로 대응되는 센서 데이터를 압축할 수 있다(430).
- [0114] 일 실시예에서, 프로세서(110)는 제1 세트의 센서 데이터들의 주기를 통일하기 위해서, 제1 세트의 센서 데이터들을 각각 사전 결정된 헤르츠(Hz)로 압축할 수 있다.
- [0115] 일 실시예에서, 프로세서(110)는 제1 세트의 센서 데이터들 중에서 센서별로 각각 하나의 대표 센서 데이터를 결정할 수 있다.
- [0116] 구체적으로, 프로세서(110)는 수치형 센서 데이터를 생성하는 제2 센서에서 획득된 복수의 제2 센서 데이터들 중에서, 복수의 제2 센서 데이터들의 평균 값에 대응되는 제2-1 센서 데이터 또는 평균 값을 상기 제2 센서의 대표 센서 데이터로 결정할 수 있다.
- [0117] 또한, 프로세서(110)는 범주형(categorical) 센서 데이터를 생성하는 제3 센서에서 획득된 복수의 제3 센서 데

이터들 중에서, 복수의 제3 센서 데이터들의 최빈값(mode)에 대응되는 제3-1 센서 데이터 또는 최빈값을 상기 제3 센서의 대표 센서 데이터로 결정할 수 있다.

[0118] 본 개시에서, 범주형 센서 데이터는 수치가 아니라, 특정 범주 또는 클래스(class)로 분류된 데이터를 의미할 수 있다. 예를 들어, 범주형 센서 데이터는 빨강, 파랑, 초록 등의 색상을 나타내는 데이터일 수 있다. 범주형 센서 데이터를 생성하는 제3 센서는 예를 들어, 색상 감지 센서 등을 포함할 수 있다.

[0119] 본 개시에서, 최빈값은 데이터 집합에서 가장 많은 빈도로 나타나는 값을 의미할 수 있다. 예를 들어, 데이터 집합에서 빨강이 3개, 파랑이 1개, 초록이 1개인 경우, 최빈값은 빨강일 수 있다.

[0120] 일 실시예에서, 프로세서(110)는 센서별로 각각 결정된 대표 센서 데이터를 사전 결정된 헤르츠로 압축할 수 있다. 예를 들어, 제1 센서가 100 헤르츠로 센서 데이터를 수집하고, 제2 센서가 50 헤르츠로 센서 데이터를 수집하는 경우, 프로세서(110)는 제1 센서 및 제2 센서의 센서 데이터를 10 헤르츠로 다운 샘플링 할 수 있다. 구체적으로, 프로세서(110)는 제1 센서의 센서 데이터를 10 헤르츠로 다운 샘플링하기 위해, 제1 센서의 원본 센서 데이터에서 매 10번째 샘플을 선택할 수 있다. 프로세서(110)는 제2 센서의 센서 데이터를 10 헤르츠로 다운 샘플링하기 위해, 제2 센서의 원본 센서 데이터에서 매 5번째 샘플을 선택할 수 있다. 프로세서(110)는 상술한 다운 샘플링을 통해 제1 센서 및 제2 센서의 데이터를 동일한 주기로 통일함으로써, 데이터 간의 비교 및 분석을 원활하게 수행할 수 있다.

[0121] 프로세서(110)는 사전 결정된 특징 선택 알고리즘을 이용하여, 센서 데이터의 변화량 및 압축된 센서 데이터로부터 적어도 하나의 특징(feature)을 선정할 수 있다(440).

[0122] 일 실시예에서, 특징 선택 알고리즘은 복수의 데이터들 중에서 중요한 것으로 판단되는 특징을 선택하는 알고리즘일 수 있다. 특징 선택 알고리즘은 인공지능 기반의 모델을 포함할 수 있다. 특징 선택 알고리즘은 그레디언트 부스팅(gradient boosting) 알고리즘, 로지스틱 회귀(logistic regression) 알고리즘 등을 포함할 수 있다. 특징 선택 알고리즘은 도 2 등에서 설명된 다른 인공지능 기반의 모델을 포함할 수도 있다.

[0123] 일 실시예에서, 그레디언트 부스팅은 이전 모델의 오차를 보완하는 방식으로 복수의 모델들이 순차적으로 연결되도록 구축되고, 최종 모델에서 예측된 값을 최종 값으로 결정할 수 있다. 그레디언트 부스팅은 센서 데이터의 변화량 및 압축된 센서 데이터를 입력으로 하고, 적어도 하나의 특징의 중요도를 출력으로 하도록 동작될 수 있다. 이러한 동작을 구현하기 위하여, 그레디언트 부스팅은 대응되는 센서의 다른 센서 데이터를 포함하는 학습 데이터셋을 이용하여 학습될 수 있다.

[0124] 일 실시예에서, 로지스틱 회귀 알고리즘은 독립 변수와 종속 변수 간의 관계를 모델링하여, 종속 변수가 특정 클래스에 속할 확률을 예측할 수 있다. 로지스틱 회귀 알고리즘은 센서 데이터의 변화량 및 압축된 센서 데이터를 입력으로 하고, 적어도 하나의 특징의 중요도를 출력으로 하도록 동작될 수 있다. 여기서, 중요도는 로지스틱 회귀 알고리즘의 계수에 기초하여 산출되는 값일 수 있다. 로지스틱 회귀 알고리즘에서 특징의 계수는 각 특징이 결과 변수에 미치는 영향을 나타낼 수 있다. 따라서, 특징의 계수 값이 클수록 해당 특징이 종속 변수에 미치는 영향이 크므로, 로지스틱 회귀 알고리즘에서 특징의 계수 값이 클수록 이에 비례하여 특징의 중요도가 클 수 있다.

[0125] 일 실시예에서, 프로세서(110)는 사전 결정된 특징 선택 알고리즘이 복수인 경우, 그레디언트 부스팅 알고리즘을 포함하는 제1 특징 선택 알고리즘을 이용하여, 센서 데이터의 변화량 및 압축된 센서 데이터로부터 적어도 하나의 제1 특징을 선정할 수 있다.

[0126] 일 실시예에서, 프로세서(110)는 로지스틱 회귀 알고리즘을 포함하는 제2 특징 선택 알고리즘을 이용하여, 센서 데이터의 변화량 및 압축된 센서 데이터로부터 적어도 하나의 제2 특징을 선정할 수 있다.

[0127] 일 실시예에서, 프로세서(110)는 적어도 하나의 제1 특징과 적어도 하나의 제2 특징을 비교함으로써, 서로 중복되는 특징인 적어도 하나의 공통 특징을 선정할 수 있다. 프로세서(110)는 상이한 특징 선택 알고리즘들을 이용하여 공통 특징을 선정함으로써, 하나의 특징 선택 알고리즘만을 이용하는 경우보다 더 정밀하게 특징을 선정할 수 있다. 따라서, 프로세서(110)는 정밀하게 선정된 특징을 이용하여 합성 데이터를 생성하고 이상 탐지 모델을 학습시킴으로써, 이상 탐지 모델의 정확도를 향상시킬 수 있다.

[0128] 몇몇 실시예에서, 프로세서(110)는 복수의 센서 데이터들 중에서 결측 값에 대해서, 시간을 기준으로 직전 혹은 직후에 발생한 값으로 채워 넣는 기법(Forward, Backward Fill)을 적용할 수 있다. 프로세서(110)는 데이터별로 정상 및 이상 여부에 대해서 레이블링을 수행할 수 있다. 따라서, 복수의 센서 데이터들은 각각 정상 또는 비정

상(이상)으로 레이블링 되어 있을 수 있다.

- [0129] 다시 도 3을 참조하면, 프로세서(110)는 인공지능 기반의 사전 학습된 데이터 증강 모델을 이용하여, 전처리된 복수의 센서 데이터들로부터 복수의 합성 데이터들을 생성할 수 있다(330).
- [0130] 일 실시예에서, 데이터 증강 모델은 기존 데이터를 변형하여 새로운 데이터 샘플을 생성하는 모델을 의미할 수 있다. 예를 들어, 데이터 증강 모델은 디퓨전 모델을 포함할 수 있다. 데이터 증강 모델은 도 2 등에서 설명된 다른 인공지능 기반의 모델을 포함할 수도 있다.
- [0131] 일 실시예에서, 디퓨전 모델은 도 2를 참조하여 상술한 바와 같이 학습된 정규 분포로부터 생성된 노이즈를 점진적으로 제거함으로써 입력 데이터와 유사한 확률 분포를 가진 결과 이미지를 생성하는 모델을 의미할 수 있다. 디퓨전 모델에 대한 구체적인 설명은 도 6을 참조하여 후술한다.
- [0132] 일 실시예에서, 데이터 증강 모델은 전처리된 상기 복수의 센서 데이터들의 각각에 정규 분포로부터 생성된 노이즈를 추가하는 확산 프로세스(diffusion process)를 수행하고 그리고 추가된 상기 노이즈를 제거하는 역확산 프로세스(reverse diffusion process)를 수행함으로써, 전처리된 상기 복수의 센서 데이터들과 유사한 상기 복수의 합성 데이터들을 생성할 수 있다.
- [0133] 프로세서(110)는 전처리된 복수의 센서 데이터들 및 복수의 합성 데이터들을 이용하여, 인공지능 기반의 이상 탐지 모델을 학습시킬 수 있다(340).
- [0134] 일 실시예에서, 이상 탐지 모델은 데이터에서 비정상적이거나 예외적인 패턴을 식별하기 위한 모델일 수 있다. 예를 들어, 이상 탐지 모델은 오토인코더를 포함할 수 있다.
- [0135] 일 실시예에서, 오토인코더는 입력 데이터와 유사한 출력 데이터를 출력하기 위한 인공지능 기반의 모델의 일종일 수 있다. 오토인코더는 적어도 하나의 히든 레이어를 포함할 수 있으며, 홀수 개의 히든 레이어가 입출력 레이어 사이에 배치될 수 있다. 각각의 레이어의 노드의 수는 입력 레이어의 노드의 수에서 병목 레이어(인코딩)라는 중간 레이어로 축소되었다가, 병목 레이어에서 출력 레이어(입력 레이어와 대칭)로 축소와 대칭되어 확장될 수도 있다. 오토인코더는 비선형 차원 감소를 수행할 수 있다. 입력 레이어 및 출력 레이어의 수는 입력 데이터의 전처리 이후에 차원과 대응될 수 있다. 오토인코더 구조에서 인코더에 포함된 히든 레이어의 노드의 수는 입력 레이어에서 멀어질수록 감소하는 구조를 가질 수 있다. 병목 레이어(인코더와 디코더 사이에 위치하는 가장 적은 노드를 가진 레이어)의 노드의 수는 너무 작은 경우 충분한 양의 정보가 전달되지 않을 수 있으므로, 특정 수 이상(예를 들어, 입력 레이어의 절반 이상 등)으로 유지될 수도 있다.
- [0136] 일 실시예에서, 오토인코더는 메모리-증강 오토인코더(Memory-Augmented Autoencoder)를 포함할 수 있다. 메모리-증강 오토인코더는 오토인코더의 기본 구조에 메모리 모듈을 추가하여, 데이터의 장기적인 패턴과 관계를 학습하는 모델일 수 있다. 메모리-증강 오토인코더에 대한 구체적인 설명은 도 7을 참조하여 후술한다.
- [0137] 일 실시예에서, 이상 탐지 모델은 전처리된 상기 복수의 센서 데이터들 및 복수의 합성 데이터들을 포함하는 학습 데이터와, 학습 데이터를 입력 받음으로써 출력되는 출력 데이터 간의 손실 값을 줄이는 방향으로 학습될 수 있다.
- [0138] 도 5는 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 센서별로 대표 센서 데이터를 결정하는 과정을 나타낸 도면이다.
- [0139] 도 5를 참조하면, 컴퓨팅 장치(100)의 프로세서(110)는 제1 시간(t_s)과 제2 시간(t_{s+1}) 사이의 구간을 제1 시간 구간(510)으로 설정할 수 있다.
- [0140] 일 실시예에서, 프로세서(110)는 제1 시간 구간의 제1 세트의 센서 데이터들 중에서 센서별로 각각 하나의 대표 센서 데이터를 결정할 수 있다.
- [0141] 예를 들어, A센서가 범주형 센서 데이터를 생성하는 센서인 경우, 프로세서(110)는 A센서에서 획득된 복수의 센서 데이터들 중에서 최빈값에 대응되는 데이터를 A센서의 대표 센서 데이터(520)로 결정할 수 있다.
- [0142] 다른 예를 들어, B센서가 수치형 센서 데이터를 생성하는 센서인 경우, 프로세서(110)는 B센서에서 획득된 복수의 센서 데이터들 중에서 평균 값에 대응되는 데이터를 B센서의 대표 센서 데이터(530)로 결정할 수 있다.
- [0143] 또 다른 예를 들어, C센서가 수치형 센서 데이터를 생성하는 센서인 경우, 프로세서(110)는 C센서에서 획득된 복수의 센서 데이터들 중에서 평균 값에 대응되는 데이터를 C센서의 대표 센서 데이터(540)로 결정할 수 있다.

- [0144] 도 6은 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 데이터 증강 모델을 이용하여 합성 데이터를 생성하는 과정을 나타낸 도면이다.
- [0145] 도 6을 참조하면, 컴퓨팅 장치(100)의 프로세서(110)는 학습 데이터(예를 들어, 전처리된 복수의 센서 데이터들 등)(610)를 데이터 증강 모델(예를 들어, 디퓨전 모델 등)(620)에 입력함으로써, 합성 데이터(640)를 생성할 수 있다.
- [0146] 데이터 증강 모델(620)에서 x_0 는 학습 데이터(610) 또는 만들어진 합성 데이터(640)에 해당할 수 있다. x_T 는 가우시안 노이즈(630)에 해당할 수 있다. $q(x_t | x_{t-1})$ 은 x_{t-1} 에 노이즈(noise)를 부여하여 x_t 를 만드는 과정일 수 있다. 해당 과정은 확산 프로세스에 해당할 수 있다. 여기서 노이즈는 가우시안 분포(Gaussian distribution)에서 선택한 임의의 벡터일 수 있다. $p_\theta(x_{t-1} | x_t)$ 는 x_t 로부터 노이즈를 제거하여 x_{t-1} 를 복원하는 과정일 수 있다. 해당 과정은 역확산 프로세스에 해당할 수 있으며, 데이터 증강 모델(예를 들어, 디퓨전 모델 등)(620)의 학습 대상일 수 있다.
- [0147] 일 실시예에서, 프로세서(110)는 학습 데이터(610)에 확산 프로세스를 여러 번 수행함으로써, 가우시안 노이즈(630)를 획득할 수 있다. 프로세서(110)는 획득된 가우시안 노이즈(630)에 역확산 프로세스를 여러 번 수행함으로써, 합성 데이터(640)를 획득할 수 있다. 상술한 과정으로 획득되는 합성 데이터(640)는 학습 데이터(610)와 유사할 수 있다.
- [0148] 일 실시예에서, 데이터 증강 모델(예를 들어, 디퓨전 모델 등)(620)의 학습 목표는 최적의 $p_\theta(x_{t-1} | x_t)$ 를 찾는 것일 수 있다. 데이터 증강 모델을 학습시키기 위한 손실함수는 아래의 수학적 식 6에 대응될 수 있다. 다만, 손실함수는 이에 한정되지 않고, 모델의 구조에 따라 변경될 수 있다.

수학적 식 6

[0149]
$$\mathbb{E}_q[D_{KL}(q(x_T | x_0) || p(x_T)) + \sum_{t=2}^T D_{KL}(q(x_{t-1} | x_t, x_0) || p_\theta(x_{t-1} | x_t)) - \log p_\theta(x_0 | x_1)]$$

- [0150] 수학적 식 6에서 $D_{KL}(\cdot)$ 는 쿨백-라이블러 발산(Kullback-Leibler divergence)을 의미할 수 있다. 일 실시예에서, 쿨백-라이블러 발산은 두 확률 분포 간의 차이를 나타내는 값을 의미할 수 있다. 쿨백-라이블러 발산은 두 확률 분포 간의 평균적인 정보량인 교차 엔트로피(cross-entropy)에서 기준이 되는 확률 분포의 엔트로피(entropy)를 차감한 값일 수 있다.
- [0151] 도 7은 본 개시의 일 실시예에 따른 컴퓨팅 장치에서 수행되는, 이상 탐지 모델을 학습시키는 과정을 나타낸 도면이다.
- [0152] 도 7을 참조하면, 컴퓨팅 장치(100)의 프로세서(110)는 학습 데이터(예를 들어, 전처리된 복수의 센서 데이터들 등)(710) 및/또는 합성 데이터(720)를 이상 탐지 모델(예를 들어, 메모리-증강 오토인코더)(730)에 입력함으로써, 손실 값(740)을 생성할 수 있다.
- [0153] 일 실시예에서, 메모리-증강 오토인코더(730)는 인코더(731), 메모리 모듈(733), 디코더(735) 등을 포함할 수 있다. 메모리-증강 오토인코더(730)의 구성들 중에서 앞서 상술한 내용과 중복되는 내용은 생략되며, 구체적인 설명은 앞서 도 2 내지 도 6을 참조하여 상술한 내용으로 대체될 수 있다.
- [0154] 일 실시예에서, 인코더(731)는 학습 데이터(710) 및/또는 합성 데이터(720)를 입력으로 받고, 잠재변수 A (732)를 출력할 수 있다.
- [0155] 일 실시예에서, 메모리 모듈(733)은 인코더(731)와 디코더(735) 사이에 위치하며 내부에 벡터 형태의 적어도 하나의 메모리 아이템(memory item)을 저장할 수 있다. 적어도 하나의 메모리 아이템은 학습될 수 있으며, 메모리-증강 오토 인코더(730)가 학습될 때마다 같이 변경될 수 있다.
- [0156] 일 실시예에서, 메모리 모듈(733)은 인코더(731)에서 출력된 잠재변수 A (732)를 전달받을 수 있다. 메모리 모듈

(733)은 잠재변수 A(732)와 저장된 메모리 아이템을 이용하여 어텐션(attention) 값을 계산할 수 있다. 메모리 모듈(733)은 계산된 어텐션 값이 사전 결정된 축소 임계 값(shrinkage threshold)보다 큰 경우, 어텐션 값 그대로 sparsity regularizer를 생성할 수 있다. 메모리 모듈(733)은 계산된 어텐션 값이 사전 결정된 축소 임계 값(shrinkage threshold)보다 작은 경우, 0으로 변경하여 sparsity regularizer를 생성할 수 있다. sparsity regularizer는 각 메모리 아이템에 대한 일종의 가중치 값일 수 있다. 메모리 모듈(733)은 각 sparsity regularizer와 메모리 아이템을 곱한 뒤 모두 더해(linear combination) 잠재변수 B(734)를 생성할 수 있다

[0157] 일 실시예에서, 디코더(735)는 생성된 잠재변수 B(734)를 입력 받고, 데이터를 재구성(reconstruction)함으로써, 재구성된 데이터를 출력할 수 있다.

[0158] 프로세서(110)는 인코더(731)에 입력된 데이터(학습 데이터(710) 및/또는 합성 데이터(720))와 디코더(735)에서 출력된 데이터를 이용하여, 손실 값(740)을 계산할 수 있다. 예를 들어, 프로세서(110)는 손실 함수를 이용하여 손실 값(740)을 계산할 수 있다.

[0159] 일 실시예에서, 손실 함수는 재구성 오차(Reconstruction Error)를 계산하는 수식 및 엔트로피 오차(Entropy error)를 계산하는 수식을 조합한 수식일 수 있다.

[0160] 재구성 오차는 인코더(731)에 입력된 데이터(학습 데이터(710) 및/또는 합성 데이터(720))와 디코더(735)에서 출력된 데이터 간의 차이를 측정한 값일 수 있다. 재구성 오차는 평균 제곱 오차(Mean Squared Error, MSE)를 사용하여 계산될 수 있다. 평균 제곱 오차는 아래의 수학적 식 7에 대응될 수 있다.

수학적 식 7

$$R(x^t, \hat{x}^t) = \frac{1}{2} \|x^t - \hat{x}^t\|^2$$

[0161]

[0162] 수학적 식 7에서 x^t 는 입력 벡터(인코더(731)에 입력된 데이터), $\hat{x}^t$ 는 출력 벡터(디코더(735)에서 출력된 데이터)일 수 있다.

[0163] 엔트로피 오차(Entropy error)는 메모리-증강 오토인코더(730)에서 사용되는 sparsity regularizer $\hat{w}$ 의 엔트로피를 계산함으로써 산출될 수 있다. Sparsity regularizer는 상술한 어텐션 값으로부터 계산되는 스칼라 값일 수 있다. 엔트로피는 이상 탐지 모델의 학습 과정에서 손실 값과 마찬가지로 값의 크기가 낮아져야 한다. sparsity regularizer $\hat{w}$ 의 엔트로피는 아래의 수학적 식 8에 대응될 수 있다.

수학적 식 8

$$E(\hat{w}^t) = -\sum_{i=1}^T \hat{w}_i \cdot \log(\hat{w}_i)$$

[0164]

[0165] 손실 함수는 수학적 식 7 및 수학적 식 8을 조합하여 생성될 수 있다. 손실 함수는 아래의 수학적 식 9에 대응될 수 있다.

수학적 식 9

$$L(\theta_e, \theta_d, M) = \frac{1}{T} \sum_{t=1}^T (R(x^t, \hat{x}^t)) + \alpha E(\hat{w}^t)$$

[0166]

- [0167] 수학식 9에서 α 는 임의의 상수, θ_e 는 인코더(731)의 파라미터, θ_d 는 디코더(735)의 파라미터, M은 메모리 모듈(733)일 수 있다.
- [0168] 일 실시예에서, 프로세서(110)는 손실 함수에서 계산된 손실 값이 최소가 되도록 이상 탐지 모델을 학습시킬 수 있다.
- [0169] 일 실시예에서, 프로세서(110)는 이상 탐지 모델에 특정 데이터를 입력하였을 때 계산된 손실 값과 사전 결정된 임계 값을 비교함으로써, 특정 데이터 또는 특정 데이터를 획득한 센서를 정상 또는 비정상(이상)으로 판별할 수 있다. 예를 들어, 프로세서(110)는 손실 값이 사전 결정된 임계 값(예를 들어, 0.5 등)보다 작으면 특정 데이터 또는 특정 데이터를 획득한 센서를 정상으로 판별할 수 있다. 다른 예를 들어, 프로세서(110)는 손실 값이 사전 결정된 임계 값보다 크면 특정 데이터 또는 특정 데이터를 획득한 센서를 비정상(이상)으로 판별할 수 있다.
- [0170] 일 실시예에서, 사전 결정된 임계 값은 해당 모델이 적용되는 도메인에 따라 다를 수 있으며, 실험을 통해서 결정될 수 있다.
- [0171] 도 8은 본 개시의 실시예들이 구현될 수 있는 예시적인 컴퓨팅 환경에 대한 간략하고 일반적인 개략도를 도시한다.
- [0172] 본 개시가 일반적으로 컴퓨팅 장치에 의해 구현될 수 있는 것으로 전술되었지만, 당업자라면 본 개시가 하나 이상의 컴퓨터 상에서 실행될 수 있는 컴퓨터 실행가능 명령어 및/또는 기타 프로그램 모듈들과 결합되어 및/또는 하드웨어와 소프트웨어의 조합으로써 구현될 수 있다는 것을 잘 알 것이다.
- [0173] 일반적으로, 프로그램 모듈은 특정의 태스크를 수행하거나 특정의 추상 데이터 유형을 구현하는 루틴, 프로그램, 컴포넌트, 데이터 구조, 기타 등등을 포함한다. 또한, 당업자라면 본 개시의 방법이 단일-프로세서 또는 멀티프로세서 컴퓨터 시스템, 미니컴퓨터, 메인프레임 컴퓨터는 물론 퍼스널 컴퓨터, 핸드헬드(handheld) 컴퓨팅 장치, 마이크로프로세서-기반 또는 프로그램가능 가전 제품, 기타 등등(이들 각각은 하나 이상의 연관된 장치와 연결되어 동작할 수 있음)을 비롯한 다른 컴퓨터 시스템 구성으로 실시될 수 있다는 것을 잘 알 것이다.
- [0174] 본 개시의 설명된 실시예들은 또한 어떤 태스크들이 통신 네트워크를 통해 연결되어 있는 원격 처리 장치들에 의해 수행되는 분산 컴퓨팅 환경에서 실시될 수 있다. 분산 컴퓨팅 환경에서, 프로그램 모듈은 로컬 및 원격 메모리 저장 장치 둘 다에 위치할 수 있다.
- [0175] 컴퓨터는 통상적으로 다양한 컴퓨터 판독가능 매체를 포함한다. 컴퓨터에 의해 액세스 가능한 매체는 그 어떤 것이든지 컴퓨터 판독가능 매체가 될 수 있고, 이러한 컴퓨터 판독가능 매체는 휘발성 및 비휘발성 매체, 일시적(transitory) 및 비일시적(non-transitory) 매체, 이동식 및 비-이동식 매체를 포함한다. 제한이 아닌 예로서, 컴퓨터 판독가능 매체는 컴퓨터 판독가능 저장 매체 및 컴퓨터 판독가능 전송 매체를 포함할 수 있다. 컴퓨터 판독가능 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보를 저장하는 임의의 방법 또는 기술로 구현되는 휘발성 및 비휘발성 매체, 일시적 및 비-일시적 매체, 이동식 및 비이동식 매체를 포함한다. 컴퓨터 판독가능 저장 매체는 RAM, ROM, EEPROM, 플래시 메모리 또는 기타 메모리 기술, CD-ROM, DVD(digital video disk) 또는 기타 광 디스크 저장 장치, 자기 카세트, 자기 테이프, 자기 디스크 저장 장치 또는 기타 자기 저장 장치, 또는 컴퓨터에 의해 액세스될 수 있고 원하는 정보를 저장하는 데 사용될 수 있는 임의의 기타 매체를 포함하지만, 이에 한정되지 않는다.
- [0176] 컴퓨터 판독가능 전송 매체는 통상적으로 반송파(carrier wave) 또는 기타 전송 메커니즘(transport mechanism)과 같은 피변조 데이터 신호(modulated data signal)에 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터 등을 구현하고 모든 정보 전달 매체를 포함한다. 피변조 데이터 신호라는 용어는 신호 내에 정보를 인코딩하도록 그 신호의 특성들 중 하나 이상을 설정 또는 변경시킨 신호를 의미한다. 제한이 아닌 예로서, 컴퓨터 판독가능 전송 매체는 유선 네트워크 또는 직접 배선 접속(direct-wired connection)과 같은 유선 매체, 그리고 음향, RF, 적외선, 기타 무선 매체와 같은 무선 매체를 포함한다. 상술된 매체들 중 임의의 것의 조합도 역시 컴퓨터 판독가능 전송 매체의 범위 안에 포함되는 것으로 한다.
- [0177] 컴퓨터(1102)를 포함하는 본 개시의 여러가지 측면들을 구현하는 예시적인 환경이 나타내어져 있으며, 컴퓨터(1102)는 처리 장치(1104), 시스템 메모리(1106) 및 시스템 버스(1108)를 포함한다. 시스템 버스(1108)는 시스템 메모리(1106)(이에 한정되지 않음)를 비롯한 시스템 컴포넌트들을 처리 장치(1104)에 연결시킨다. 처리 장치(1104)는 다양한 상용 프로세서들 중 임의의 프로세서일 수 있다. 듀얼 프로세서 및 기타 멀티프로세서 아키텍

처도 역시 처리 장치(1104)로서 이용될 수 있다.

- [0178] 시스템 버스(1108)는 메모리 버스, 주변장치 버스, 및 다양한 상용 버스 아키텍처 중 임의의 것을 사용하는 로컬 버스에 추가적으로 상호 연결될 수 있는 몇 가지 유형의 버스 구조 중 임의의 것일 수 있다. 시스템 메모리(1106)는 판독 전용 메모리(ROM)(1110) 및 랜덤 액세스 메모리(RAM)(1112)를 포함한다. 기본 입/출력 시스템(BIOS)은 ROM, EPROM, EEPROM 등의 비휘발성 메모리(1110)에 저장되며, 이 BIOS는 시동 중과 같은 때에 컴퓨터(1102) 내의 구성요소들 간에 정보를 전송하는 일을 돕는 기본적인 루틴을 포함한다. RAM(1112)은 또한 데이터를 캐싱하기 위한 정적 RAM 등의 고속 RAM을 포함할 수 있다.
- [0179] 컴퓨터(1102)는 또한 내장형 하드 디스크 드라이브(HDD)(1114)(예를 들어, EIDE, SATA) -이 내장형 하드 디스크 드라이브(1114)는 또한 적당한 새시(도시 생략) 내에서 외장형 용도로 구성될 수 있음-, 자기 플로피 디스크 드라이브(FDD)(1116)(예를 들어, 이동식 디스켓(1118)으로부터 판독을 하거나 그에 기록을 하기 위한 것임), 및 광 디스크 드라이브(1120)(예를 들어, CD-ROM 디스크(1122)를 판독하거나 DVD 등의 기타 고용량 광 매체로부터 판독을 하거나 그에 기록을 하기 위한 것임)를 포함한다. 하드 디스크 드라이브(1114), 자기 디스크 드라이브(1116) 및 광 디스크 드라이브(1120)는 각각 하드 디스크 드라이브 인터페이스(1124), 자기 디스크 드라이브 인터페이스(1126) 및 광 드라이브 인터페이스(1128)에 의해 시스템 버스(1108)에 연결될 수 있다. 외장형 드라이브 구현을 위한 인터페이스(1124)는 USB(Universal Serial Bus) 및 IEEE 1394 인터페이스 기술 중 적어도 하나 또는 그 둘 다를 포함한다.
- [0180] 이들 드라이브 및 그와 연관된 컴퓨터 판독가능 매체는 데이터, 데이터 구조, 컴퓨터 실행가능 명령어, 기타 등등의 비휘발성 저장을 제공한다. 컴퓨터(1102)의 경우, 드라이브 및 매체는 임의의 데이터를 적당한 디지털 형식으로 저장하는 것에 대응한다. 상기에서의 컴퓨터 판독가능 매체에 대한 설명이 HDD, 이동식 자기 디스크, 및 CD 또는 DVD 등의 이동식 광 매체를 언급하고 있지만, 당업자라면 zip 드라이브(zip drive), 자기 카세트, 플래쉬 메모리 카드, 카트리지, 기타 등등의 컴퓨터에 의해 판독가능한 다른 유형의 매체도 역시 예시적인 운영 환경에서 사용될 수 있으며 또 임의의 이러한 매체가 본 개시의 방법들을 수행하기 위한 컴퓨터 실행가능 명령어를 포함할 수 있다는 것을 잘 알 것이다.
- [0181] 운영 체제(1130), 하나 이상의 애플리케이션 프로그램(1132), 기타 프로그램 모듈(1134) 및 프로그램 데이터(1136)를 비롯한 다수의 프로그램 모듈이 드라이브 및 RAM(1112)에 저장될 수 있다. 운영 체제, 애플리케이션, 모듈 및/또는 데이터의 전부 또는 그 일부가 또한 RAM(1112)에 캐싱될 수 있다. 본 개시가 여러가지 상업적으로 이용가능한 운영 체제 또는 운영 체제들의 조합에서 구현될 수 있다는 것을 잘 알 것이다.
- [0182] 사용자는 하나 이상의 유선/무선 입력 장치, 예를 들어, 키보드(1138) 및 마우스(1140) 등의 포인팅 장치를 통해 컴퓨터(1102)에 명령 및 정보를 입력할 수 있다. 기타 입력 장치(도시 생략)로는 마이크, IR 리모콘, 조이스틱, 게임 패드, 스타일러스 펜, 터치 스크린, 기타 등등이 있을 수 있다. 이들 및 기타 입력 장치가 종종 시스템 버스(1108)에 연결되어 있는 입력 장치 인터페이스(1142)를 통해 처리 장치(1104)에 연결되지만, 병렬 포트, IEEE 1394 직렬 포트, 게임 포트, USB 포트, IR 인터페이스, 기타 등등의 기타 인터페이스에 의해 연결될 수 있다.
- [0183] 모니터(1144) 또는 다른 유형의 디스플레이 장치도 역시 비디오 어댑터(1146) 등의 인터페이스를 통해 시스템 버스(1108)에 연결된다. 모니터(1144)에 부가하여, 컴퓨터는 일반적으로 스피커, 프린터, 기타 등등의 기타 주변 출력 장치(도시 생략)를 포함한다.
- [0184] 컴퓨터(1102)는 유선 및/또는 무선 통신을 통한 원격 컴퓨터(들)(1148) 등의 하나 이상의 원격 컴퓨터로의 논리적 연결을 사용하여 네트워크화된 환경에서 동작할 수 있다. 원격 컴퓨터(들)(1148)는 워크스테이션, 컴퓨팅 디바이스 컴퓨터, 라우터, 퍼스널 컴퓨터, 휴대용 컴퓨터, 마이크로프로세서-기반 오락 기기, 피어 장치 또는 기타 통상의 네트워크 노드일 수 있으며, 일반적으로 컴퓨터(1102)에 대해 기술된 구성요소들 중 다수 또는 그 전부를 포함하지만, 간략함을 위해, 메모리 저장 장치(1150)만이 도시되어 있다. 도시되어 있는 논리적 연결은 근거리 통신망(LAN)(1152) 및/또는 더 큰 네트워크, 예를 들어, 원거리 통신망(WAN)(1154)에의 유선/무선 연결을 포함한다. 이러한 LAN 및 WAN 네트워킹 환경은 사무실 및 회사에서 일반적인 것이며, 인트라넷 등의 전사적 컴퓨터 네트워크(enterprise-wide computer network)를 용이하게 해주며, 이들 모두는 전세계 컴퓨터 네트워크, 예를 들어, 인터넷에 연결될 수 있다.
- [0185] LAN 네트워킹 환경에서 사용될 때, 컴퓨터(1102)는 유선 및/또는 무선 통신 네트워크 인터페이스 또는 어댑터(1156)를 통해 로컬 네트워크(1152)에 연결된다. 어댑터(1156)는 LAN(1152)에의 유선 또는 무선 통신을 용이하

게 해줄 수 있으며, 이 LAN(1152)은 또한 무선 어댑터(1156)와 통신하기 위해 그에 설치되어 있는 무선 액세스 포인트를 포함하고 있다. WAN 네트워킹 환경에서 사용될 때, 컴퓨터(1102)는 모뎀(1158)을 포함할 수 있거나, WAN(1154) 상의 통신 컴퓨팅 디바이스에 연결되거나, 또는 인터넷을 통하는 등, WAN(1154)을 통해 통신을 설정하는 기타 수단을 갖는다. 내장형 또는 외장형 및 유선 또는 무선 장치일 수 있는 모뎀(1158)은 직렬 포트 인터페이스(1142)를 통해 시스템 버스(1108)에 연결된다. 네트워킹화된 환경에서, 컴퓨터(1102)에 대해 설명된 프로그램 모듈들 또는 그의 일부분이 원격 메모리/저장 장치(1150)에 저장될 수 있다. 도시된 네트워크 연결이 예시적인 것이며 컴퓨터들 사이에 통신 링크를 설정하는 기타 수단이 사용될 수 있다는 것을 잘 알 것이다.

[0186] 컴퓨터(1102)는 무선 통신으로 배치되어 동작하는 임의의 무선 장치 또는 개체, 예를 들어, 프린터, 스캐너, 데스크톱 및/또는 휴대용 컴퓨터, PDA(portable data assistant), 통신 위성, 무선 검출가능 태그와 연관된 임의의 장비 또는 장소, 및 전화와 통신을 하는 동작을 한다. 이것은 적어도 Wi-Fi 및 블루투스 무선 기술을 포함한다. 따라서, 통신은 종래의 네트워크에서와 같이 미리 정의된 구조이거나 단순히 적어도 2개의 장치 사이의 애드혹 통신(ad hoc communication)일 수 있다.

[0187] Wi-Fi(Wireless Fidelity)는 유선 없이도 인터넷 등으로의 연결을 가능하게 해준다. Wi-Fi는 이러한 장치, 예를 들어, 컴퓨터가 실내에서 및 실외에서, 즉 기지국의 통화권 내의 아무 곳에서나 데이터를 전송 및 수신할 수 있게 해주는 셀 전화와 같은 무선 기술이다. Wi-Fi 네트워크는 안전하고 신뢰성 있으며 고속인 무선 연결을 제공하기 위해 IEEE 802.11(a, b, g, 기타)이라고 하는 무선 기술을 사용한다. 컴퓨터를 서로에, 인터넷에 및 유선 네트워크(IEEE 802.3 또는 이더넷을 사용함)에 연결시키기 위해 Wi-Fi가 사용될 수 있다. Wi-Fi 네트워크는 비인가 2.4 및 5GHz 무선 대역에서, 예를 들어, 11Mbps(802.11a) 또는 54 Mbps(802.11b) 데이터 레이트로 동작하거나, 양 대역(듀얼 대역)을 포함하는 제품에서 동작할 수 있다.

[0188] 본 개시의 기술 분야에서 통상의 지식을 가진 자는 정보 및 신호들이 임의의 다양한 상이한 기술들 및 기법들을 이용하여 표현될 수 있다는 것을 이해할 것이다. 예를 들어, 위의 설명에서 참조될 수 있는 데이터, 지시들, 명령들, 정보, 신호들, 비트들, 심볼들 및 칩들은 전압들, 전류들, 전자기파들, 자기장들 또는 입자들, 광학장들 또는 입자들, 또는 이들의 임의의 결합에 의해 표현될 수 있다.

[0189] 본 개시의 기술 분야에서 통상의 지식을 가진 자는 여기에 개시된 실시예들과 관련하여 설명된 다양한 예시적인 논리 블록들, 모듈들, 프로세서들, 수단들, 회로들 및 알고리즘 단계들이 전자 하드웨어, (편의를 위해, 여기에서 소프트웨어로 지칭되는) 다양한 형태들의 프로그램 또는 설계 코드 또는 이들 모두의 결합에 의해 구현될 수 있다는 것을 이해할 것이다. 하드웨어 및 소프트웨어의 이러한 상호 호환성을 명확하게 설명하기 위해, 다양한 예시적인 컴포넌트들, 블록들, 모듈들, 회로들 및 단계들이 이들의 기능과 관련하여 위에서 일반적으로 설명되었다. 이러한 기능이 하드웨어 또는 소프트웨어로서 구현되는지 여부는 특정한 애플리케이션 및 전체 시스템에 대하여 부과되는 설계 제약들에 따라 좌우된다. 본 개시의 기술 분야에서 통상의 지식을 가진 자는 각각의 특정한 애플리케이션에 대하여 다양한 방식으로 설명된 기능을 구현할 수 있으나, 이러한 구현 결정들은 본 개시의 범위를 벗어나는 것으로 해석되어서는 안 될 것이다.

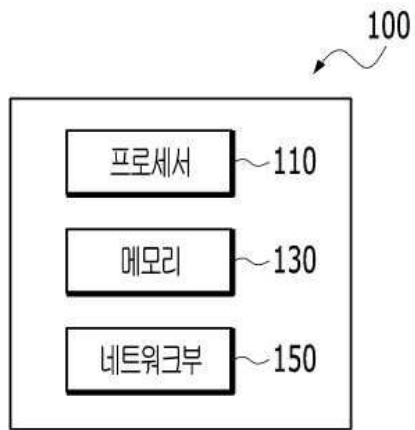
[0190] 여기서 제시된 다양한 실시예들은 방법, 장치, 또는 표준 프로그래밍 및/또는 엔지니어링 기술을 사용한 제조물품(article)으로 구현될 수 있다. 용어 제조물품은 임의의 컴퓨터-판독가능 저장장치로부터 액세스 가능한 컴퓨터 프로그램, 캐리어, 또는 매체(media)를 포함한다. 예를 들어, 컴퓨터-판독가능 저장매체는 자기 저장 장치(예를 들면, 하드 디스크, 플로피 디스크, 자기 스트립, 등), 광학 디스크(예를 들면, CD, DVD, 등), 스마트 카드, 및 플래쉬 메모리 장치(예를 들면, EEPROM, 카드, 스틱, 키 드라이브, 등)를 포함하지만, 이들로 제한되는 것은 아니다. 또한, 여기서 제시되는 다양한 저장 매체는 정보를 저장하기 위한 하나 이상의 장치 및/또는 다른 기계-판독가능한 매체를 포함한다.

[0191] 제시된 프로세스들에 있는 단계들의 특정한 순서 또는 계층 구조는 예시적인 접근들의 일례임을 이해하도록 한다. 설계 우선순위들에 기반하여, 본 개시의 범위 내에서 프로세스들에 있는 단계들의 특정한 순서 또는 계층 구조가 재배열될 수 있다는 것을 이해하도록 한다. 첨부된 방법 청구항들은 샘플 순서로 다양한 단계들의 엘리먼트들을 제공하지만 제시된 특정한 순서 또는 계층 구조에 한정되는 것을 의미하지는 않는다.

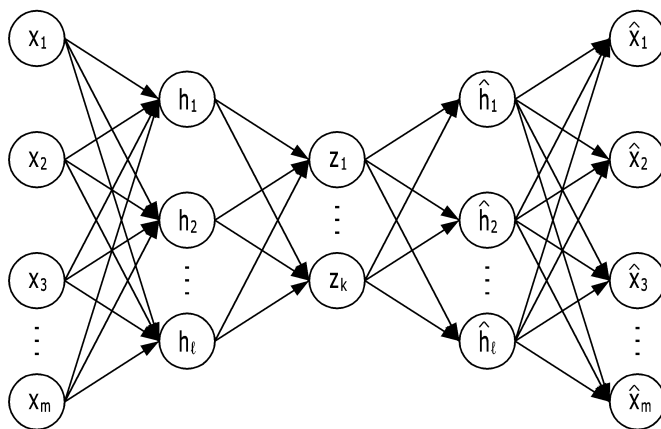
[0192] 제시된 실시예들에 대한 설명은 임의의 본 개시의 기술 분야에서 통상의 지식을 가진 자가 본 개시를 이용하거나 또는 실시할 수 있도록 제공된다. 이러한 실시예들에 대한 다양한 변형들은 본 개시의 기술 분야에서 통상의 지식을 가진 자에게 명백할 것이며, 여기에 정의된 일반적인 원리들은 본 개시의 범위를 벗어남이 없이 다른 실시예들에 적용될 수 있다. 그리하여, 본 개시는 여기에 제시된 실시예들로 한정되는 것이 아니라, 여기에 제시된 원리들 및 신규한 특징들과 일관되는 최광의의 범위에서 해석되어야 할 것이다.

도면

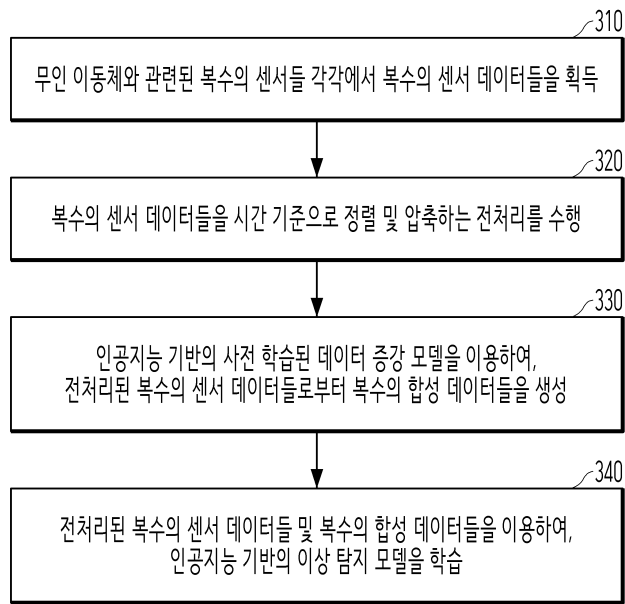
도면1



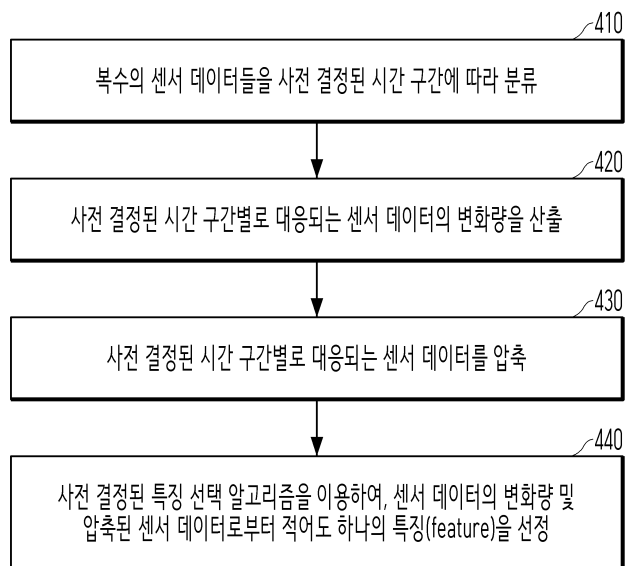
도면2



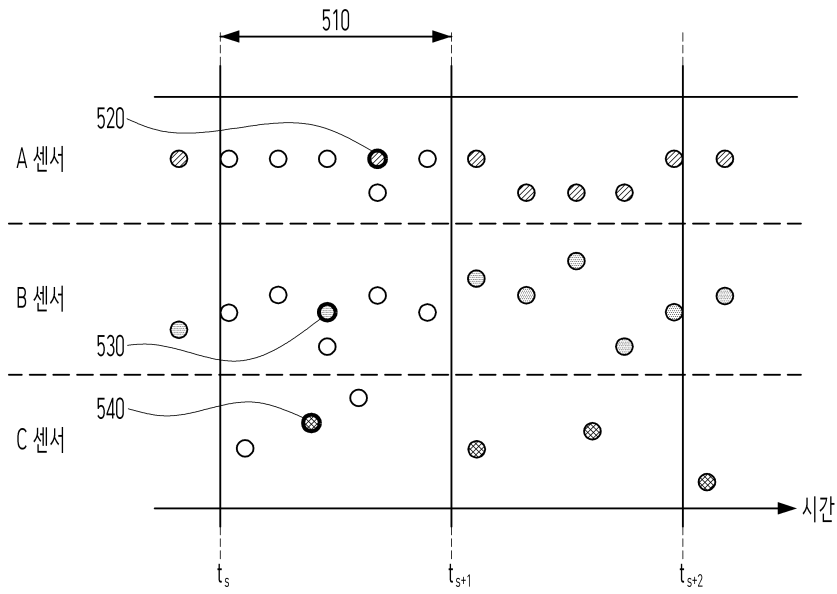
도면3



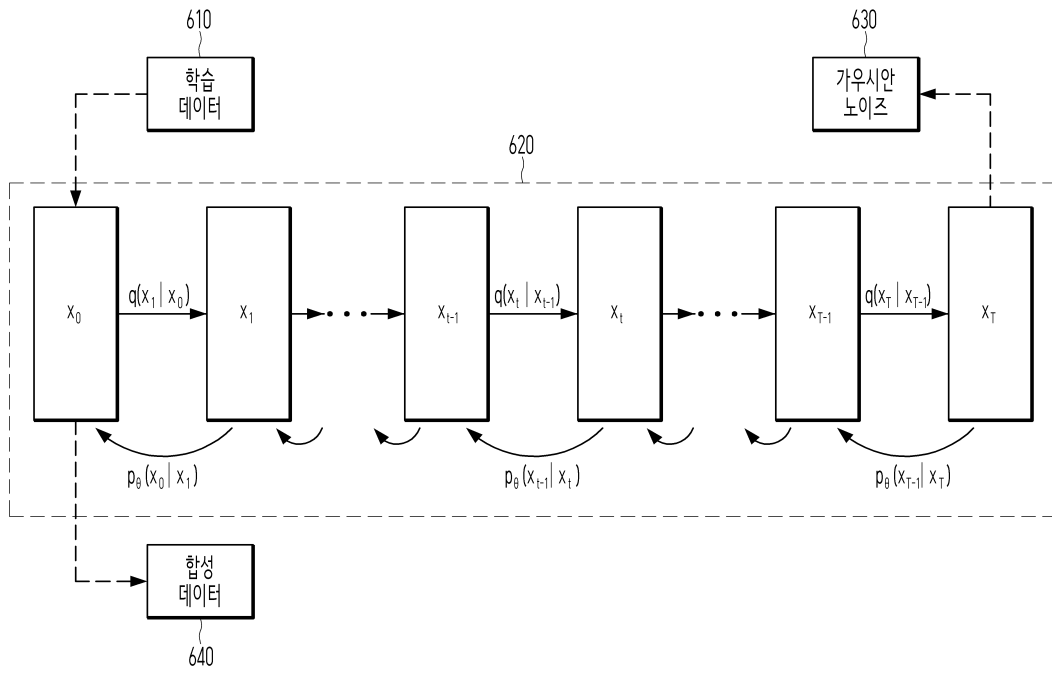
도면4



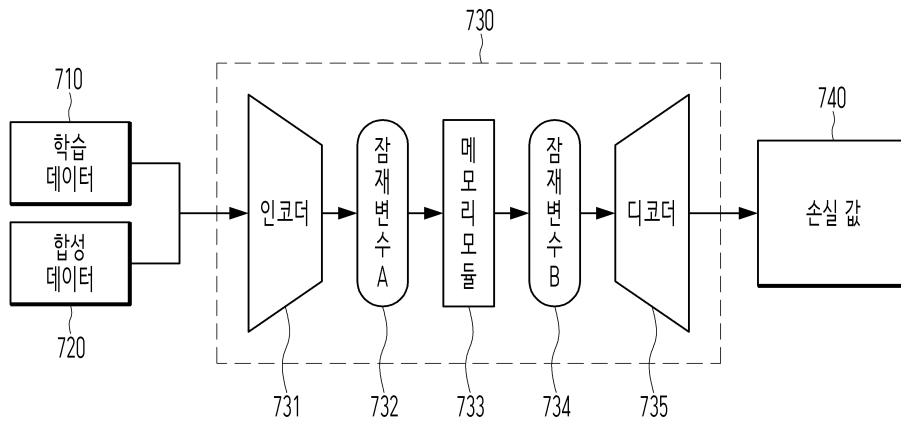
도면5



도면6



도면7



도면8

