

인공지능 정리

인공지능이 활용되는 분야

- 1) 음성 인식 기술
- 2) 자동화 및 스마트 기기
- 3) 이미지 및 비디오 인식
- 4) 금융 및 경제 분야
- 5) 의료 분야
- 6) 건강 모니터링 및 착용 기기
- 7) 약물 개발 및 생명 과학 연구
- 8) 암 예측 및 예방
- 9) 언어 처리 및 자연어 이해
- 10) 실시간 번역 및 서비스
- 11) 교육 및 학습 지원
- 12) 지능형 교통 및 이동 서비스

ChatGPT

-2022년 11월 30일 출시

-2023년 3월 14일 GPT-4출시

인공지능

- 기계가 지능을 갖도록 하는 것
- 지능은 주변 환경에서 기계가 적절하고 예측 가능한 방식으로 동작하는 것, 개념이 모호
- 다양한 기술, 소프트웨어, 컴퓨터 시스템을 포함

인간의 학습법을 모방한 인공지능

- 학습 : 경험을 통해서 얻은 결과로 패턴을 분석하는 것.
- 빅데이터를 활용하여 사람처럼 스스로 학습.

인공지능은 어떻게 작동하는가?

- 데이터 분석 : 인공지능이 많은 데이터를 분석 및 학습
- 패턴 인식 : 새로운 데이터에서 패턴을 인식
- 작업 수행 : 인식된 패턴을 통해서 새로운 작업을 수행

인공신경망 : 사람처럼 생각하는 기계의 개발

- 신경망의 수학적 모델링 : 1943년
- 생각하는 기계 : 1950년 앨런 튜링이 제안
- 인공지능이라는 용어 사용 : 1956년 존 매커시.
- 인공신경망 퍼셉트론의 개발 : 1957년

머신러닝과 딥러닝

- 머신러닝 : 특정 데이터를 입력으로 받아 자동으로 분류 등과 작업을 수행할 수 있는 프로그램을 자동으로 만들어내는 기술
- 딥러닝 : 특징 추출부터 추론 / 예측까지 전 주기의 자동화를 실현한 머신러닝의 한 분야
- 범위 : 인공지능 > 머신러닝 > 딥러닝

머신러닝

- 컴퓨터가 경험적 데이터를 학습해서 지능적으로 동작할 수 있는 기능을 갖추는 인공지능의 세부 분야
- 데이터와 결과값을 입력, 프로그램을 생산

머신러닝 학습 방식

- 감독 학습(지도 학습) : 데이터(x)와 사람이 작업을 한 라벨(y)를 합쳐 학습 데이터(x, y)를 구축. 학습데이터를 통해 x에 대한 y의 관계를 매핑한 예측 모델 생성 및 활용. 주유 태스크는 분류와 회귀.
- 비감독 학습 : 라벨 없이 단순히 데이터에 나타나는 통상적인 유사, 공통 현상을 학습. 인간의 학습의 원형으로 간주. 주요 태스크는 클러스터링, 밀도 예측
- 강화학습 : 최근에 각광받는 기술. 어떤 상태에 장기적 보상을 최대화하는 동작을 매핑하는 정책을 산출.

머신러닝 기법의 구성

- 학습데이터, 모델, 학습 알고리즘의 3요소로 구성.
- 모델은 매우 다양, 단순하게는 파라미터의 집합으로 추상화될 수 있다. 모델로 예측할 때 결정경계의 모양으로 선형, 비선형으로 분류한다.
- 학습 알고리즘 : 모델을 학습하는 알고리즘, 머신러닝 기법에 따라 달라짐

딥러닝의 부상 이유

- 딥러닝은 새로운 기술이 아니다.
- 시대적 환경(컴퓨팅 환경 + 풍부한 학습 데이터 + 딥러닝 알고리즘)의 산물이다.

딥러닝 특징

- 다층 구조. 추출 특징 수 증대.
- 계층적 추상화 학습
- End-to-End 학습. 딥러닝은 Raw data를 입력으로 특징 자동추출 -> 전문가 개입 X.

인공지능이 어떻게 학습하는가?

- 뇌에서 일어나는 신경망의 동작을 자연스럽게 구현하기는 어렵다.
- 사람의 뇌가 학습하는 동작을 그대로 모방
- 신경망 : 뇌의 구조를 모방한 정보 처리와 기억하는 방식을 구현

인공신경망(Artificial Neural Network)란?

- 사람의 생물학적 신경망을 모방한 인공지능 기술
- 연결주의의 대표적 인공지능 기술
- 안정적인 학습 기능, 병렬처리와 에러 감내 등의 장점
- 초기부터 음성, 영상, 자연어 처리 등에 많이 응용되고 있음.

인공신경망의 고안

- AND게이트, OR게이트, NOT게이트를 구현할 수 있다는 이론을 제시
- 존 폰 노이만이 고안

인공지능의 발전사

- 1950년대에 인공지능의 1차 전성기
- 기호주의 : 인간의 지능과 지식을 기호화, 매뉴얼화하여 컴퓨터가 문제를 해결 체스, 미로, 퍼즐 등 논리로 해결할 수 있는 문제. 실현 가능하고 인공지능 기술의 기초.
- 퍼셉트론(Perceptron), 연결주의 : 뇌의 뉴런이 서로 연결되어 정보를 처리하는 구조를 모방한 알고리즘, 인공신경망의 기초. 연구의 트렌드가 기호주의에서 연결주의로 옮겨짐
- 1969년 마빈 민스키가 퍼셉트론의 한계를 수학적으로 증명, AI 빙하기 시작.

신경망의 수학적 모델링

- 사람 뇌의 신경망의 동작을 모방
- 뉴런의 정보 전달 방식을 모방, 사람의 신경망을 흉내내는 방식을 고안.

생물학적 신경망(biological network)

- 뉴런 : 뇌의 기본 단위로 작용
- 시냅스 : 뉴런 간 신호 전달을 담당
- 사람의 뇌 : 1000억 개 이상 뉴런, 100조 개의 시냅스

생물학적 신경망의 학습

- 입력 : 외부 자극이나 정보가 뇌에 도착, 전기적 신호로 변환되어 뇌로 전송
- 처리 : 입력된 정보가 뇌의 신경망을 통해 전파. 뉴런 간의 시냅스를 통해 전달되는 신호는 강도와 패턴에 따라 다름. 이 패턴은 학습과 기억에 중요한 역할.
- 출력 : 처리된 정보는 다시 전기적 신호로 변환되어 다른 뉴런들을 통해 목적지로 전달.
- 학습과 기억 : 경험을 통해 학습과 기억, 시냅스의 강도가 변화함으로써 새로운 경험을 학습, 이 학습이 지속되면 장기 기억으로 옮겨짐.
- 숙달된 신경망은 시냅스가 적음. 불필요한 시냅스 연결 부위를 제거하고 시냅스 연결 부위는 강화되는 연결망 최적화를 하기 때문. -> 아이보다 어른의 시냅스가 더 적음.

인공신경망의 뉴런 모델

- 생물학적 뉴런 : 뉴런에서 입력 신호 처리, 시냅스에서 뉴런 간의 신호를 전달, 시냅스의 연결 강도가 학습됨
- 인공 뉴런 네트워크 : 뉴런을 노드, 신호 전달은 엷지로 표현. 엷지의 가중치 크기를 조절하여 시냅스를 표현. 학습은 가중치를 변경하여 구현.

퍼셉트론

- 1957년, 프랑크 로젠블라트가 고안한 알고리즘. 단순한 인공 신경망 구조. 뉴런을 모방하여 작동. 입력 신호에 가중치를 부여. 활성화 함수를 통해 처리

퍼셉트론의 연산식

- 퍼셉트론 : 다수의 신호를 입력 받아서 하나의 신호를 출력
- $y = w_1x_1 + w_2x_2$, 임계값 세타 보다 작거나 같으면 0, 크면 1
- 편향 도입 시 $y = w_1x_1 + w_2x_2 + b$ 가 0보다 작거나 같으면 0, 크면 1. $b = -\text{세타}$

퍼셉트론의 구조와 동작

- 입력층 : 단순히 데이터를 입력하는 역할, 뉴런은 아님
- 출력층 : 일반적으로 한 개의 뉴런으로 구성, 여러 개의 출력 뉴런도 가능하지만 퍼셉트론의 특징 분석에는 한 개의 뉴런으로 충분.
- 뉴런 : 활성화 함수로 시그넘 함수의 수정본 사용. 임계치를 포함한 입력가중합의 계산. 출력은 1 또는 -1.

결정 경계

- 결정 경계에 의한 이진 분류
- 1) 변수 1개 입력 : 점
- 2) 변수 2개 입력 : 직선
- 3) 변수 3개 입력 : 평면

선형 분리성

- 단지 성형 분리 가능한 데이터만 분류 가능. 결정경계가 선형함수이기 때문. 퍼셉트론이 선형분리 가능한 데이터만 분리 가능하다는 것은 퍼셉트론의 한계로 작용.
- AND, OR, NOT, AND등은 직선으로 분리가능하기 때문에 퍼셉트론으로 구현 가능.
- XOR 연산은 선형분리가 불가. 구현도 불가능.

분석 기반 학습

- 퍼셉트론 학습은 적합한 가중치를 찾는 것.
- 분석 기반 학습 : 간단한 경우 수식을 풀어 가중치를 구하는 방식

퍼셉트론 학습 알고리즘 개요

- Rosenblatt, 1960년
- 학습 데이터셋 준비, 가중치 초기값 설정
- 각 입력에 대해 오차가 감소하도록 가중치 조정
- 모든 입력이 정확하게 분류될 때까지 여러 번 반복

퍼셉트론 학습 알고리즘의 동작 구조

- 학습 데이터는 N개 학습샘플
- x를 퍼셉트론에 입력, 산출되는 y와 d를 비교해서 에러를 정정
- $w_{i,t+1} = w_{i,t} + \eta(d - y)x_i$
- 에러(손실값) = $d - y$

퍼셉트론 학습 알고리즘의 수렴성

- 퍼셉트론 학습은 선형분리 가능한 데이터에 대해선 무조건 수렴
- 안정성을 나타내며 신경망에 대해서는 최초
- 선형 분리 가능하지 않은 데이터에 대해서는 발산 또는 진동

퍼셉트론의 작동 원리 : 정리

- 단일 뉴런 신경망 구조 : 입력층, 가중치, 활성화 함수, 출력층 등으로 구성된 간단한 신경망 구조를 가지고 있음
- 활성화 함수 : 입력 신호의 총합을 계산, 계단 함수, 시그모이드 함수, 렐루 함수 등을 사용

퍼셉트론의 활용

- 패턴 인식, 이진 분류, 신경망 기초

퍼셉트론 한계점

- 비선형 문제 해결 불가

퍼셉트론과 인공신경망의 관계

- 초기 : 퍼셉트론은 초기 인공 신경망 모델로 알려졌으며 단층 퍼셉트론으로 구성
- 현재 : 인공 신경망은 퍼셉트론의 한계를 극복하기 위해 다층 퍼셉트론과 같이 발전

다층 퍼셉트론

- 다층 퍼셉트론으로 XOR 게이트 해결하기
- 그래프를 휘게 접어서 네모 점과 세모 점을 한 개의 직선으로 구분할 수 있다
- 공간을 왜곡하여 문제를 해결
- 그래프를 휘게 접는 역할 추가 = 퍼셉트론에 은닉층 추가 = 다층 퍼셉트론

XOR게이트

- AND, NAND, OR게이트 필요

다층 퍼셉트론

- 여러 개의 퍼셉트론을 쌓아서 구성한 신경망
- 수많은 가중치와 효율적인 계산을 위해 벡터의 내적을 이용
- 은닉층에서 다양한 활성화 함수를 이용하여 연산

다층 퍼셉트론의 한계

- 다층 퍼셉트론만으로 모든 문제 해결 불가능 : 은닉층을 추가 할수록 역전파시에 연산량이 늘어나고 기울기 소실의 문제점 발생.
- 다양한 활성화 함수를 함께 이용하여 문제를 해결해야 함.

신경망(Neural Network)란?

- 인간의 두뇌를 모방한 연결주의의 대표적 AI 기술
- 안정적인 학습 기능, 병렬처리와 에러 감내 등의 장점

인공신경망의 학습

- 학습에 필요한 지식인 학습데이터를 입력으로 사용
- 학습과정에서 학습데이터에서 추추한 지식이 뉴런 간의 연결강도(weights)에 반영
- 학습 -> 가중치를 적합한 값으로 변경하는 작업

뉴런의 연산 모델

- 편향 : 뉴런 입력과 별도로 뉴런에 가해지는 값. 순수 가중 입력합을 변경시킨다.
- 편향은 순수 가중입력합에 어파인 변환을 적용하는 효과
- 임계치 : 뉴런의 가중입력합을 편향과 반대 방향으로 변경. 임계치는 단순히 편향의 반대 부호값

활성화 개요

- 활성화 함수 : 뉴런의 정보처리 방식을 결정. 입력가중합이 어떻게 처리되는지를 결정.
- 처리 유형 : 활성화 여부 or 활성화 정도 or 활성화 확률

선형 계열 활성화 함수

- 선형(Linear)함수 : 입력과 출력이 동일
- Step 함수 : 0을 기준으로 출력이 0이거나 1로 표현
- Squashing activation function : -1과 1사이의 값만 그대로 출력 나머지는 다 -1, 1
- 시그넘 함수 : 계단 함수 : 0을 기준으로 1, 0, -1, 0이면 1로 출력하는 것으로 변경.
- Relu 함수 : 가중합이 음수일 경우에 출력은 무조건 0. 가중합이 양수일 경우에는 상한 값이 없음. 시그모이드 함수에서 나타나는 에러 기울기 소실이나 폭증의 문제점 해소.

시그모이드 활성화 함수

- 로지스틱 시그모이드 함수 : [0, 1] 범위의 출력 값을 갖는 부드러운 비선형 함수
- 출력값이 확률인 경우에도 사용 가능. $y = 1/(1 + e^{-x})$
- 미분 가능

신경망의 종류

- 신경망의 신호의 연산을 처리하는 방향에 따라서 분류
- Feedforward구조 : 입력부에서 출력부로 단방향으로 신호 전달. Feedback 연결선 없음.
ex) 퍼셉트론, FFN, CNN
- Feedback구조 : 신호가 이전 계층으로 다시 전달. Feedback 연결선 존재. 분석 난해.
ex) RNN, LSTM, 홉필드 네트워크, 볼츠만 머신

신경망의 구조

- 다수의 계층이 횡적으로 나열된 형태, 입력층과 출력층 사이 다수의 은닉층이 존재.
- 계층은 보통 계층에 속한 뉴런과 앞 계층과의 가중치 연결선을 포함한 개념.
- 계층 간의 가중치 연결 방식의 종류 : 완전연결 vs 부분 연결
- 일반적인 뉴럴 네트워크는 완전연결층을 사용.

신경망의 동작

- 계층별 뉴런의 동기적 활성화 : 계층 간 동기화 데이터플로우 컴퓨팅 방식으로 동작.

신경망 학습 및 추론

- 학습 : 동일한 입력이라도 가중치가 다르면 출력이 달라질 수 있음.
- 추론 : 동일한 가중치 환경의 뉴럴 네트워크에서 서로 다른 입력을 가한 경우 일반적으로 출력이 달라짐.

다층신경망의 개요

- 퍼셉트론 : 1957년에 Frank Rosenblatt에 의해 제안. 입력과 가중치를 곱한 값에 활성화 함수를 적용하는 단순한 구조. 선형분리성의 한계
- 다층 퍼셉트론 : 2개의 계층으로 구성된 다층 퍼셉트론으로 XOR기능 구현 입증. 당시에는 역전파 알고리즘 등과 같이 심층 신경망을 효과적으로 훈련시킬 방법 X.
- 다층 신경망의 연혁 : 2개 이상의 계층을 갖는 구조 활용의 최대 걸림돌인 학습 문제를 오류 역전파 알고리즘으로 해결.
- 역전파 알고리즘 : 1986년에 제안. 최적화를 통해 다층 신경망을 효과적으로 훈련.
- 비선형 활성화 함수 도입
- 딥러닝 : 2000년대 중반 이후 컴퓨팅 파워의 증가와 더 효과적인 최적화 기법의 도입으로 다층 신경망에 기반한 딥러닝이 부활
- 초기에는 다층 퍼셉트론으로 불렸지만, 시그모이드 함수를 사용하는 신경망은 더 이상 퍼셉트론이 아니다.
- 일반 신경망(FNN, Feedforward Neural Network)라고도 불림.

FNN의 구조

- (1) No cyclic link (2) 한 개의 은닉층 (3) 다양한 활성화 함수 사용
- 은닉층이 2개 이상 -> Deep FNN
- 일반적으로 각 계층은 완전연결층
- 계층은 '앞 계층과의 가중치'까지 포함한 의미

뉴럴 계층

- bias : 뉴런을 얼마나 잘 활성화할 것인지를 조정하는 매개변수
- 활성화 함수 : 입력 신호를 내보낼 것인지 말지를 결정
- 출력층 함수 : 결론을 내주는 함수

가중치 행렬 표기 방식

- W12 : x2 -> y1로 가는 가중치
- W23 : x3 -> y2로 가는 가중치

비선형 뉴럴 계층

- 활성화 함수에 따라 다양한 정류가 존재
- 어파인->시그모이드, 어파인->ReLU, -> 어파인 -> 소프트맥스

소프트맥스 계층

- 소프트맥스 함수 계층은 입력벡터를 확률로 변환하는 기능을 가짐.
- 입력 : 1.2, 0.7, 2.3
- 출력 : $y1 = e^{1.2} / (e^{1.2} + e^{0.7} + e^{2.3})$
- 출력층과 통합 활용, 출력층과 독립 활용 등으로 활용됨.
- 출력의 합은 항상 1이 됨.

순방향 처리

- 신경망의 주기능. 신경망의 태스크를 수행.
- 역방향의 학습과정에서도 순방향 처리 과정에서 산출된 중간 결과가 사용됨.
- 순방향 처리의 실제 구현에서는 단일 모드와 배치모드 방식을 사용

배치 처리

- 단일 모드 : 한 번에 한 개의 입력을 처리
- 배치 모드 : 한 번에 여러 개의 입력을 함께 처리

손글씨 MNIST 분석

- Label은 이미지가 나타내는 숫자가 어떤 숫자인지를 나타내는 10개의 숫자로 이루어진 1행 행렬, 해당 숫자의 인덱스를 1, 나머지는 0.
- 훈련 데이터 셋은 이미지가 6만개. 라벨도 6만개.
- Test 데이터 셋은 이미지가 1만개. 라벨도 1만개.
- 이미지 데이터는 28x28 회색조 영상으로 구성 -> 784개의 입력층

load_mnist()

- MNIST 데이터셋 파일(mnist.pkl)을 읽어서 넘파이 배열로 만들어 출력해주는 함수
- normalize, flatten, one_hot_label 등의 flag가 있음.

MNIST 분류 신경망 (FNN) 구현

- MNIST 테스트 및 훈련 데이터의 예측 정확도 계산
- 사전에 훈련된 가중치/편향 파일(pickle 형식)을 사용
- 입력층 : 784개 노드
- 계층1(은닉층) : 100개 노드, Sigmoid 함수 사용
- 계층2(출력층) : 50개 노드, Sigmoid 함수 사용
- 소프트맥스 계층
- 동작 모델 : 샘플모드와 배치모드 각각 구현

소프트웨어 구성

- 신경망 지원 함수
 - 1) sigmoid()
 - 2) softmax()
- 신경망 모델 관련 함수
 - 1) setup_network()
 - 2) predict()
- MNIST 분류 메인 로직
 - 1) FNN 생성 및 MNIST 데이터 로딩
 - 2) 샘플 모드 MNIST 분류
 - 3) 배치 모드 MNIST 분류

배치 모드 망의 흐름

- 배치 모드로 하는 것이 일반적
- 100개의 BATCH_SIZE로 할 시 784개의 데이터를 100개 입력.
- 10개로 분류한 결과가 100개 출력.

신경망의 학습

- 가중치를 적합하게 조정하는 작업
- 분석적 기법 : 방정식을 구한 후 해를 계산하는 방식
- 에러정정 학습법 : 오차를 점진적으로 줄여가는 방식

손실함수

- 손실함수 : 목표치와 실제 출력 간의 차이(손실, 에러)를 계산하는 함수
- 평균제곱에러, 교차엔트로피 에러 등

평균제곱에러(MSE, Mean Square Error)

- 제곱에러 : 출력값과 목표치의 값이 제곱에 1/2 곱한 값 : $1/2(d-y)^2$

다중 분류(Multiple classification)

- q개의 클래스 중에서 하나를 선택하는 것
- 입력 x가 어떤 클래스에 가장 많은 확률로 포함되는지 출력

교차 엔트로피 에러(CEE, Cross Entropy Error)

- 주로 다중 분류를 위한 소프트맥스 함수의 출력에 대해 적용.
- 우도 함수 사용. 우도 함수(likelihood function)는 실현된 데이터로부터 특정 통계 모델의 적합성을 확인하는데 주로 이용

함수의 기울기

- 기울기 0 : 함수 값 불변(최소 또는 최대점)
- 기울기를 알면 최소 또는 최대값의 방향을 알 수 있음
- 접선의 기울기 = 미분값

기울기 강하 기법

- 함수의 접선 기울기를 사용해 함수의 최소값을 구함
- 정확하게는 최소값을 갖는 파라미터(독립변수 x 값)을 구함
- 시작점부터 반복적으로 파라미터 값을 갱신
- 손실함수를 통해 손실이 최소가 되는 지점을 찾음
- 현재 위치에서 손실함수의 기울기가 국소점으로 가는 방향으로 가도록 설정
- 다음 $x = \text{현재 } x - \text{갱신률 } x \text{ 기울기}$
- $0 \leq \text{학습률} \leq 1$ 의 영향 : 작으면 느린 수렴, 크면 빠르지만 최소점 주위에서 진동.
- 한계 : 전역최소점 대신 국소 최소점만을 탐색할 경우가 많음.

기울기 강하 기법의 신경망 학습 적용

- 에러 기울기는 입력샘플 x 에 대한 신경망의 손실함수에 대한 특정 가중치에 대한 접선의 기울기
- 1. 가중치 초기화
- 2. 현재의 각 가중치 값에 대해 에러 기울기 계산
- 3. 기울기 부호의 반대방향으로 가중치 값을 변경
- 4. 수렴 시까지 2, 3 반복

역전파

- 노드의 에러 기울기 : 노드의 가중입력합에 대한 에러 함수의 기울기. 최종 에러가 역전파되어 영향을 받음. 손실함수를 미분하여 가중입력합을 대입한 값.
- 가중치 에러 기울기 : 가중치에 대한 에러함수의 기울기. 손실함수를 미분하여 가중치를 대입한 값.

역전파 학습 개념

- 노드 에러 기울기를 역전파, 가중치 에러 기울기를 구하여 가중치 수정.

출력층 노드의 에러 기울기

- 손실함수를 평균제곱에러를 사용했을 때
- 미분한 값에 가중입력합 대입.

에러 기울기 구하기

1. 가중입력합 구하기
2. 출력값 구하기
3. 손실함수를 미분하여 출력 대입, 평균에러 일시 $-(d-y)$.
4. 활성화함수 미분한 값에 출력 대입, 시그모이드 일시 출력 $x(1-출력)$
5. 3번과 4번을 곱하기