

머신러닝과 딥러닝

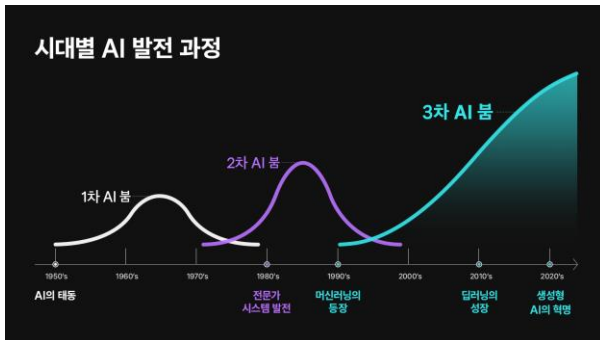
연준모

- 개요
- 딥러닝과 머신러닝
- 기본 개념 알고가기
- Colab으로 MLP 알아보기
- 정보통신과 AI

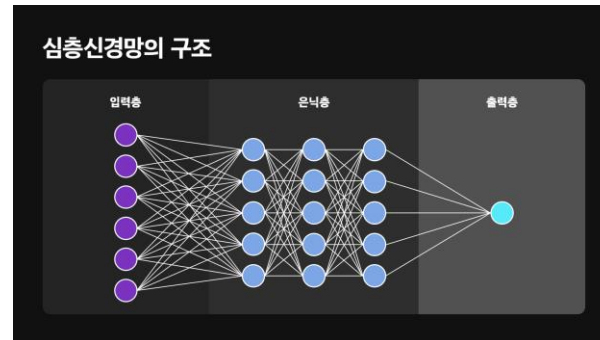
1. 개요

어떤걸 배우나요?

AI 시대의 시작



AI의 작동 원리

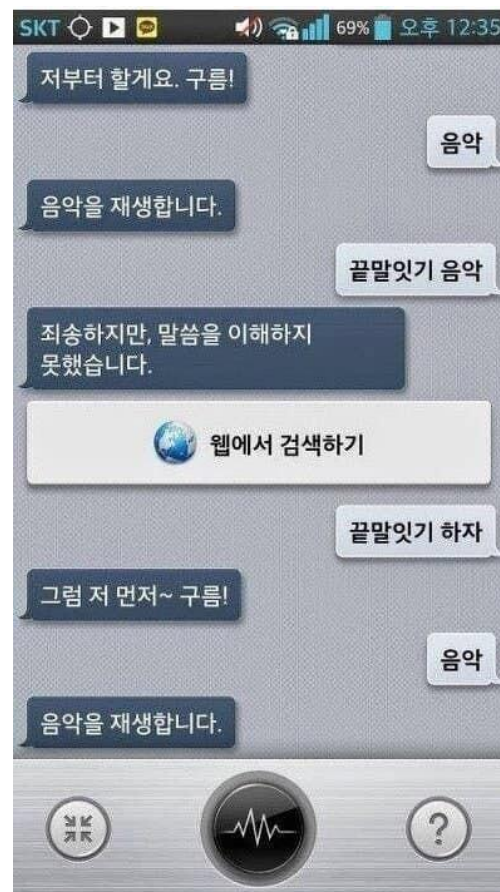
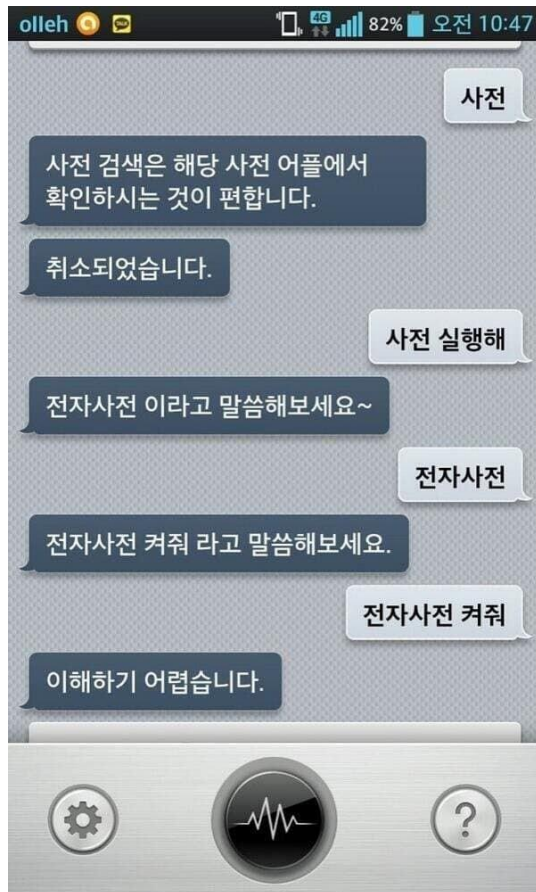


정보통신과 AI

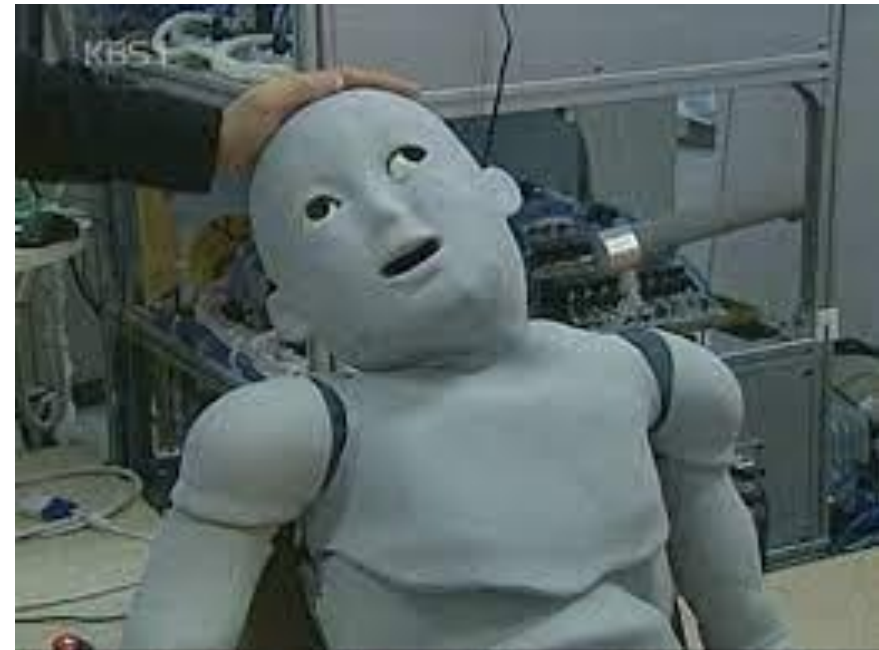


1. 개요

AI 혁신의 시작



불과 15년 전,
멍청한 모습만 보여왔던 AI

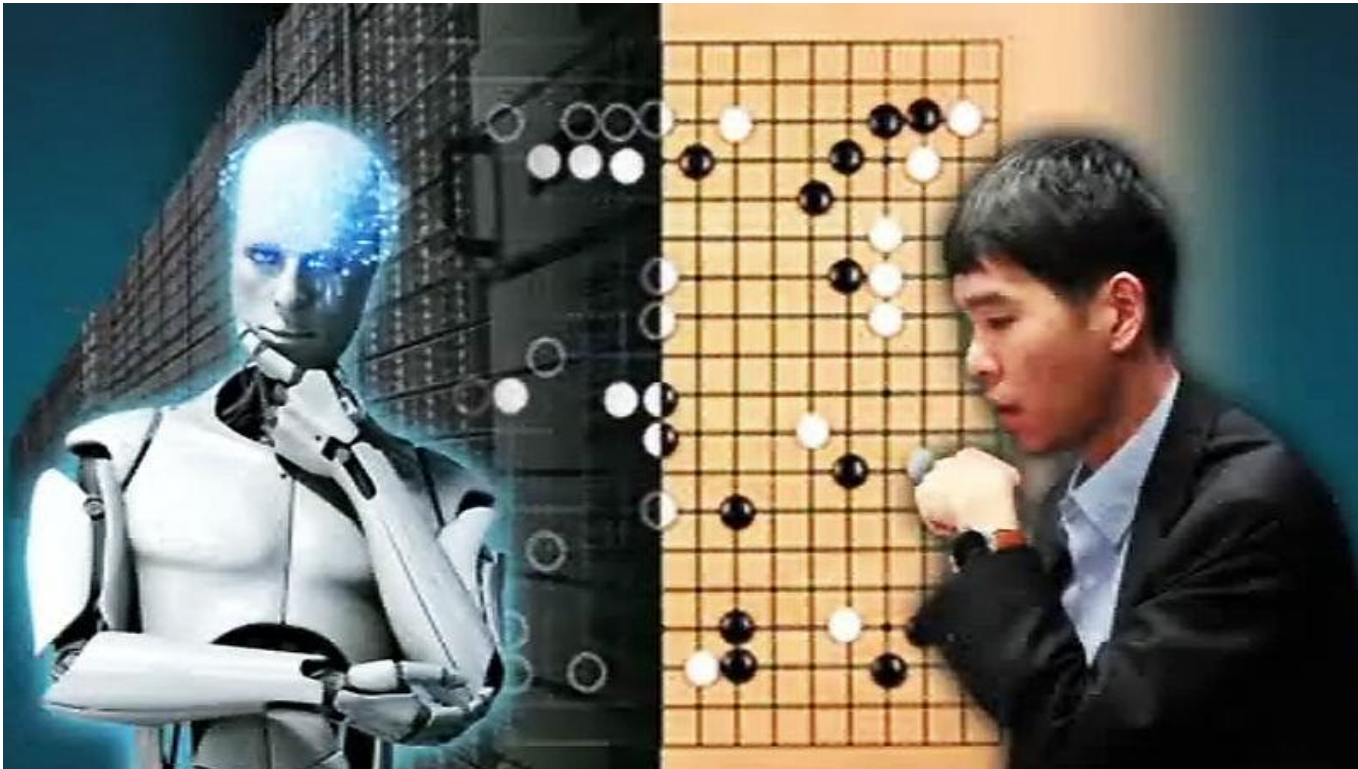


1. 개요

ONECLICK AI

AI 혁신의 시작

처음으로 똑똑하다 알려진 AI



인간의 바둑 실력을 압도하는 모습을 처음으로 보여줌!

이때가 무려 2016년

AI 혁신의 시작

2017년 전설의 논문

이 알고리즘을 기반으로 만들어진 GPT 등 기타 AI들!

The screenshot shows the top portion of an arXiv paper page. The header is dark red with the arXiv logo and navigation links. Below the header, the paper title 'Attention Is All You Need' is prominently displayed in bold black text. The authors' names are listed in blue. The abstract text is in a smaller black font. At the bottom, there are metadata fields for subjects, citation information, and a DOI link.

arXiv > cs > arXiv:1706.03762 Search... Help | Adv

Computer Science > Computation and Language

[Submitted on 12 Jun 2017 (v1), last revised 2 Aug 2023 (this version, v7)]

Attention Is All You Need

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin

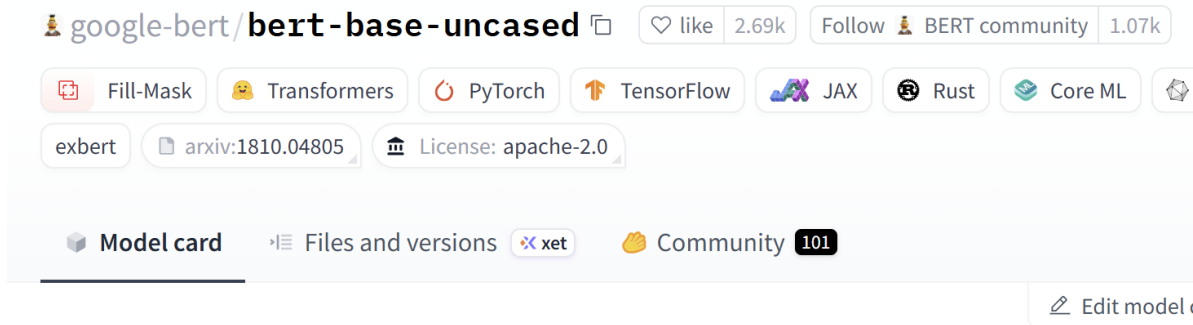
The dominant sequence transduction models are based on complex recurrent or convolutional neural networks in an encoder-decoder configuration. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

Comments: 15 pages, 5 figures

Subjects: **Computation and Language (cs.CL)**; Machine Learning (cs.LG)

Cite as: [arXiv:1706.03762](https://arxiv.org/abs/1706.03762) [cs.CL]
(or [arXiv:1706.03762v7](https://arxiv.org/abs/1706.03762v7) [cs.CL] for this version)
<https://doi.org/10.48550/arXiv.1706.03762> ⓘ

AI 혁신의 시작



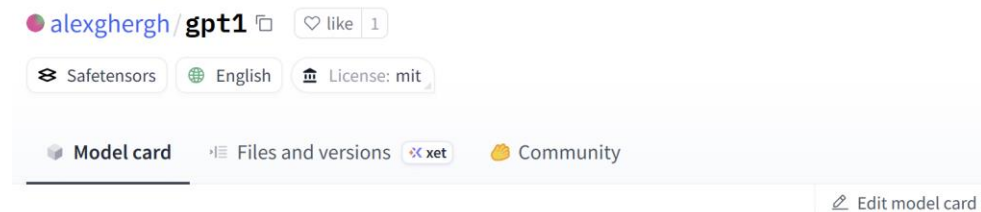
BERT base model (uncased)

Pretrained model on English language using a masked language modeling (MLM) objective. It was introduced in [this paper](#) and first released in [this repository](#). This model is uncased: it does not make a difference between english and English.

Disclaimer: The team releasing BERT did not write a model card for this model so this model card has been written by the Hugging Face team.

2018년 BERT 개발

자연어를 이해하기 시작하는 AI



Model description

This repository aims to re-create the GPT 1 architecture, using HuggingFace's `transformers`.

The original paper of the model can be found [here](#). The blog post accompanying this paper is [here](#). The code and weights can be found [here](#).

The original model was trained, as noted in OpenAI's blogpost, 1 month on 8 GPU's (P600's), on the original BookCorpus dataset (containing around ~7000 books).

1. 개요

ONECLICK AI

AI 혁신의 시작

2020년 GPT3 개발

모델의 사이즈를 본격적으로 키우기 시작!



Nvidia Tesla A100 80G
Tesla / Ampere
CUDA 6912 80G

2020년 5월 출시



엔비디아의 A100 그래픽카드가 없었다면
불가능했던 AI

1. 개요

AI 혁신의 시작

2022년 Chat GPT 출시



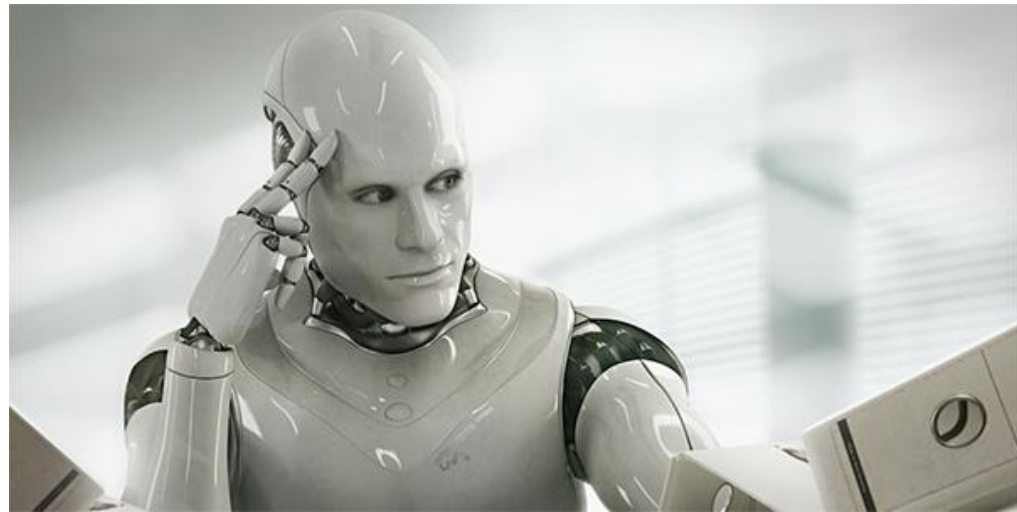
장기기억, 눈, 귀, 생각 모든것을 얻은 에이전트



과제, 공부를 도와주며 모두가 사용하게 된 AI

1. 개요

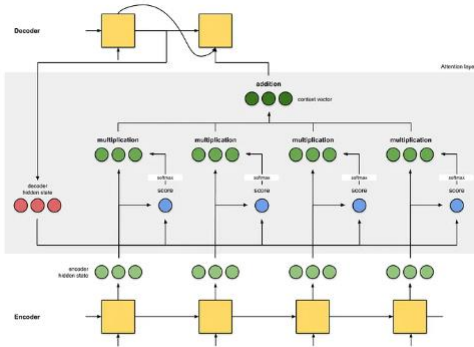
AI 혁신의 시작



그럼 대체 무엇이 AI를 똑똑하게 만들었는가?

AI 혁신의 시작

Attention



모델 크기 증가 A100

똑똑해져 개발자의 밥그릇을 위협하는 AI

다양한 딥러닝 기법 들

Lola

GGLU

RAG

1. 개요

ONECLICK AI

이 전에는 무엇이 있었는가?

알고리즘! 지금도 엄청나게 많이 쓰인다..

함수이기에, 명확한 해답과 정답이 있다!

$$Y = x + 1$$

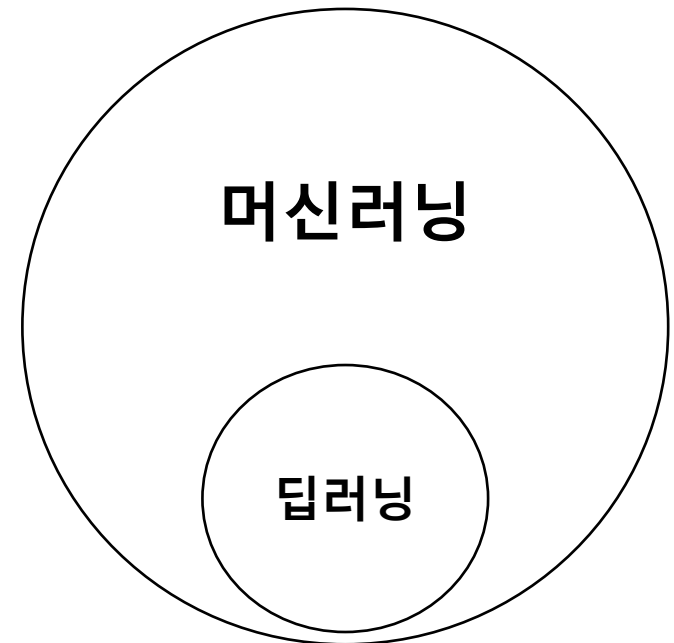
X 가 1 이면?

Y 는 2!

중요한 건 '규칙'
을 알고있다는 것!

그런데, '**규칙**' 을 모를 경우에는 어떻게 하나요?

이럴 땐 바로 '**머신러닝!**'



이 전에는 무엇이 있었는가?

'규칙' 대신 알고 있는 정답!

$X = 1, y = 2...$

$X = 5, y = 6...$

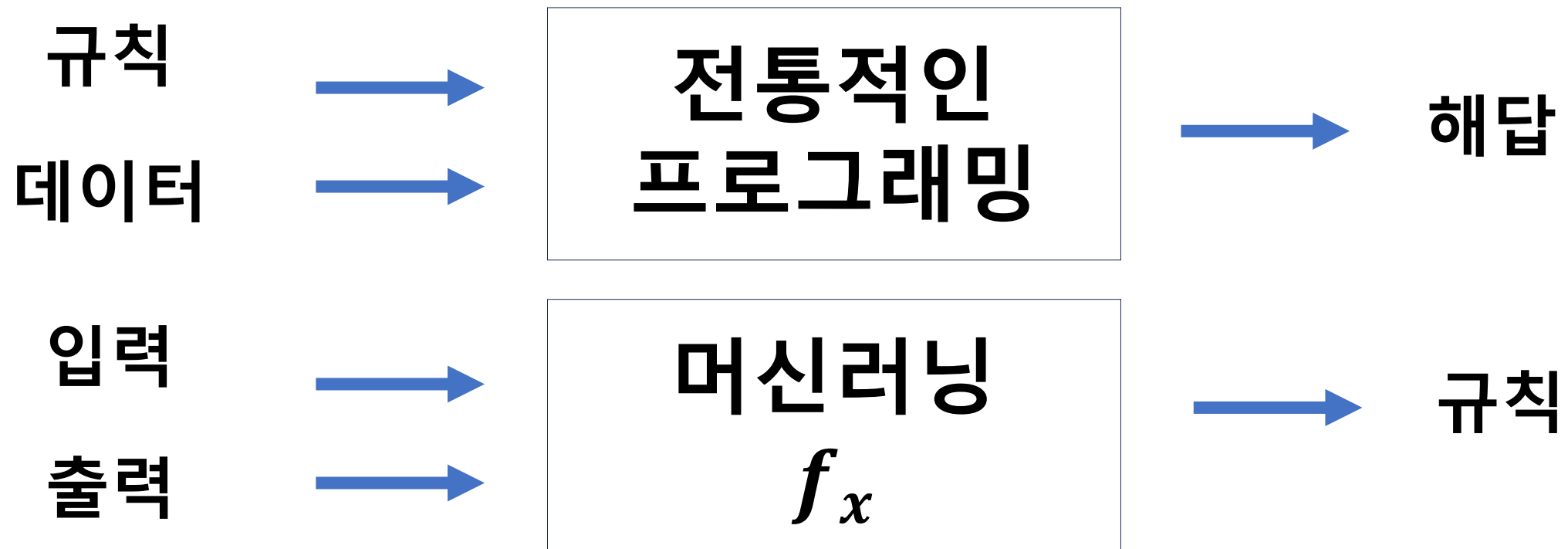
...

$X = 256, y = 257$

많은 입력, 정답 데이터로 규칙을 찾는다!

2. 딥러닝과 머신러닝 머신러닝이란?

알고리즘과 머신러닝의 차이



2. 딥러닝과 머신러닝 머신러닝과 딥러닝의 차이

ONECLICK AI

머신러닝과 딥러닝의 차이 머신러닝

자 이건
고양이야
알겠어??



이게 고양이 귀고, 이게 고양이 털이고,
이게 고양이 눈이고, 이게 고양이 코고 ...

고양이의 모든 특징을
하나 하나 알려줘야 하는 머신러닝!

AI

AI

AI

AI

AI

○ = ○ =

2. 딥러닝과 머신러닝 머신러닝과 딥러닝의 차이

ONECLICK AI

머신러닝과 딥러닝의 차이 딥러닝

이게 뭐라고?



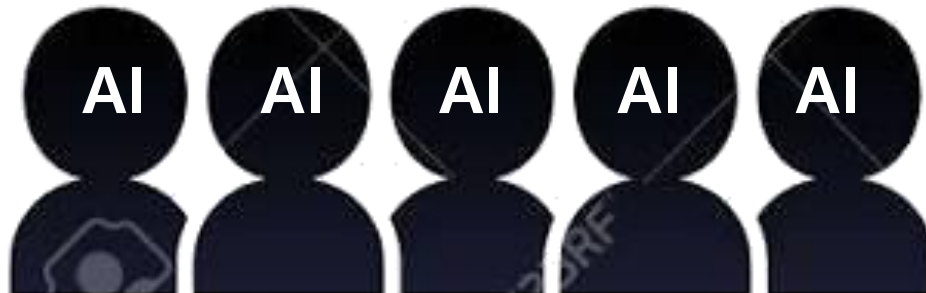
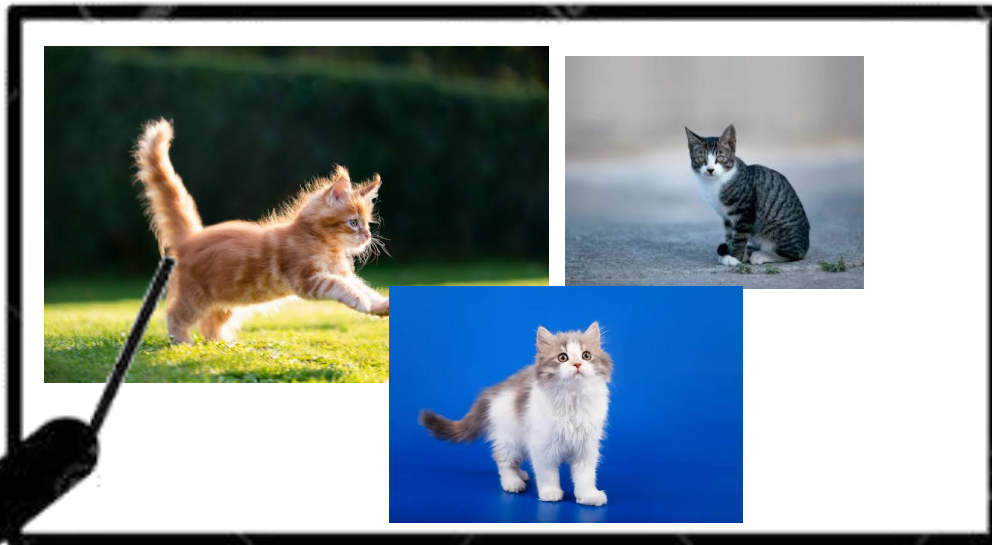
알려준 특징들을 종합해 입력 데이터가
뭔지 맞추는 머신러닝!

2. 딥러닝과 머신러닝 머신러닝과 딥러닝의 차이

ONECLICK AI

머신러닝과 딥러닝의 차이 딥러닝

자 이건
고양이야
알겠어??



특징을 알아서 추출하는 딥러닝!

이게 뭐라고?



AI

고양이!

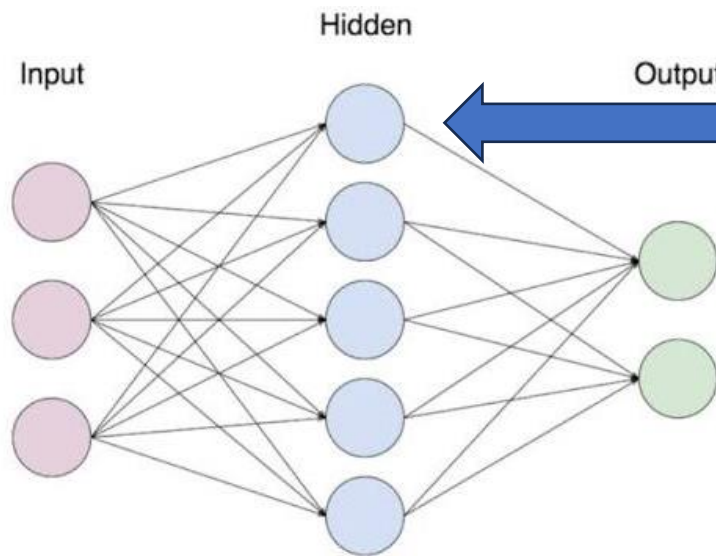
○ = ○ =

2. 딥러닝과 머신러닝 머신러닝과 딥러닝의 차이

ONECLICK AI

머신러닝과 딥러닝의 차이 딥러닝

그러면 딥러닝 모델은, 저게 고양이인지 어떻게 알고 있는걸까??



노드

각 노드에는 전부
가중치 Weight
편향 Bias
을 가지고 있다.



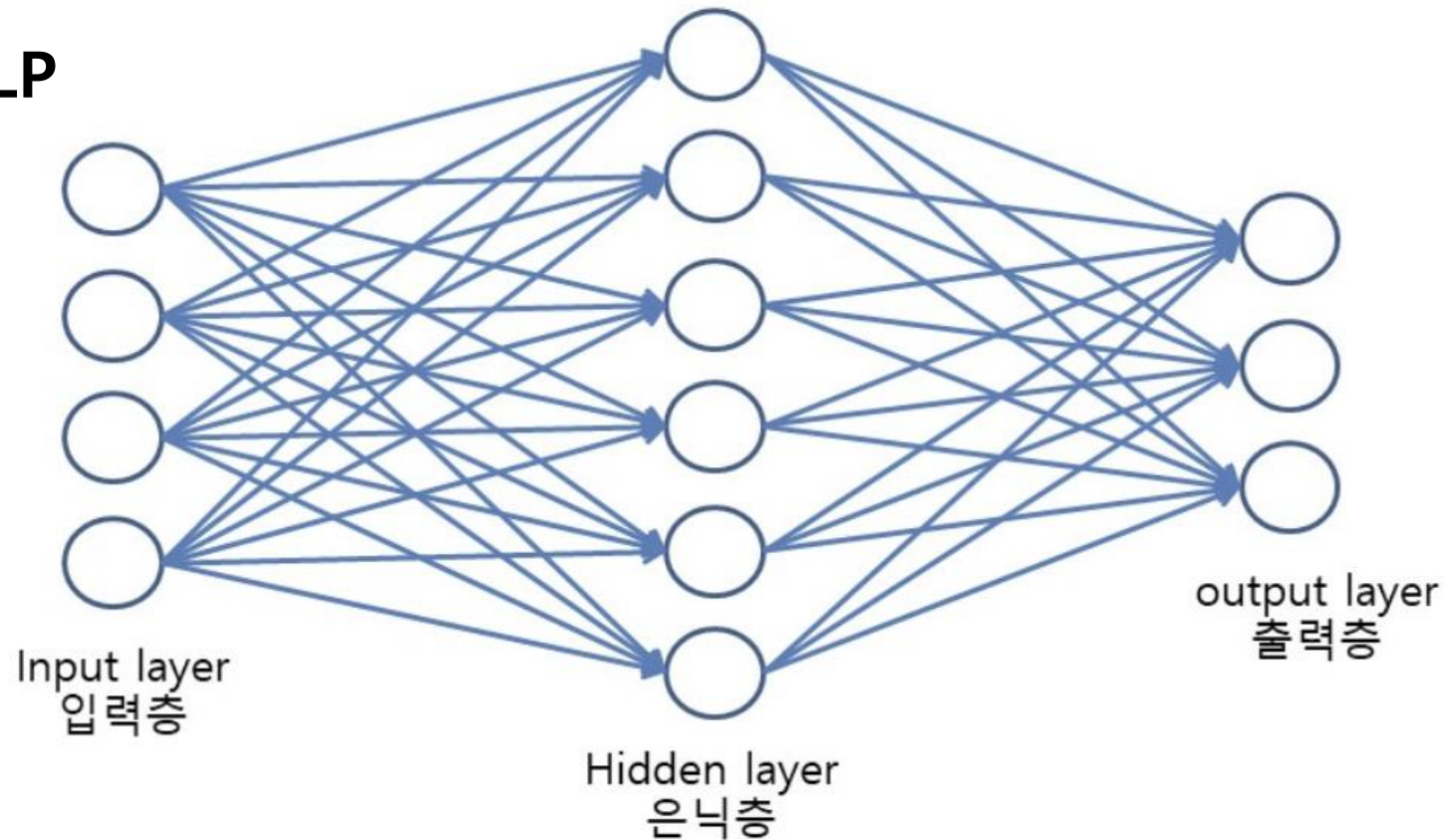
각 가중치와 편향의 값은
저게 고양이라는 사실을
기억하고 있다.

바로 모델이 정보를 저장하는 Weight, Bias 에 값이 저장되어 있기 때문!

2. 딥러닝과 머신러닝 어떻게 생겼을까?

ONECLICK AI

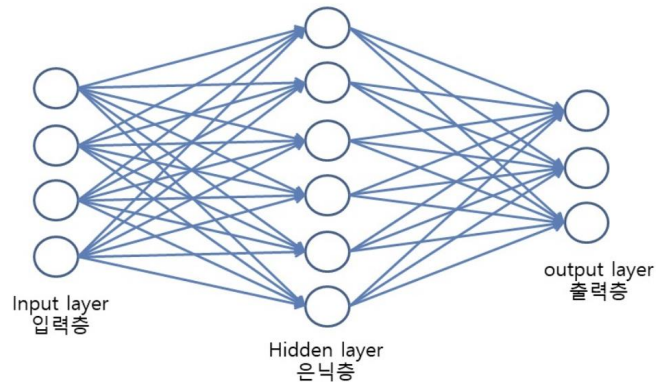
오늘 알아볼 딥러닝 MLP



올바른 가중치 값을 얻기 위해(규칙) 딥러닝 한다
딥러닝이 잘 되면, 더 정답에 가까운 가중치(규칙)를 얻게 된다.

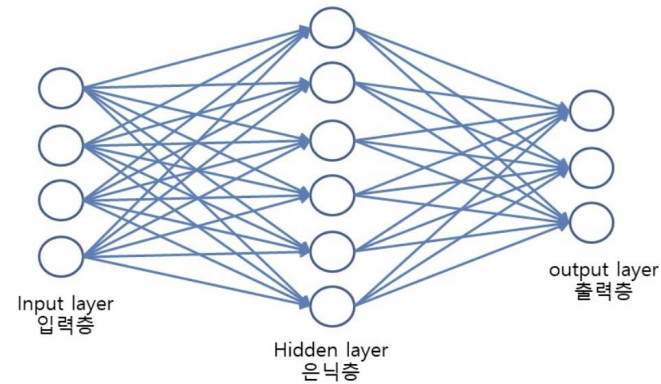
3. 기본 개념 알고가기 순전파와 역전파

순전파



입력 데이터가 신경망의 층 들을 순서대로 통과
예측값을 계산하는 과정

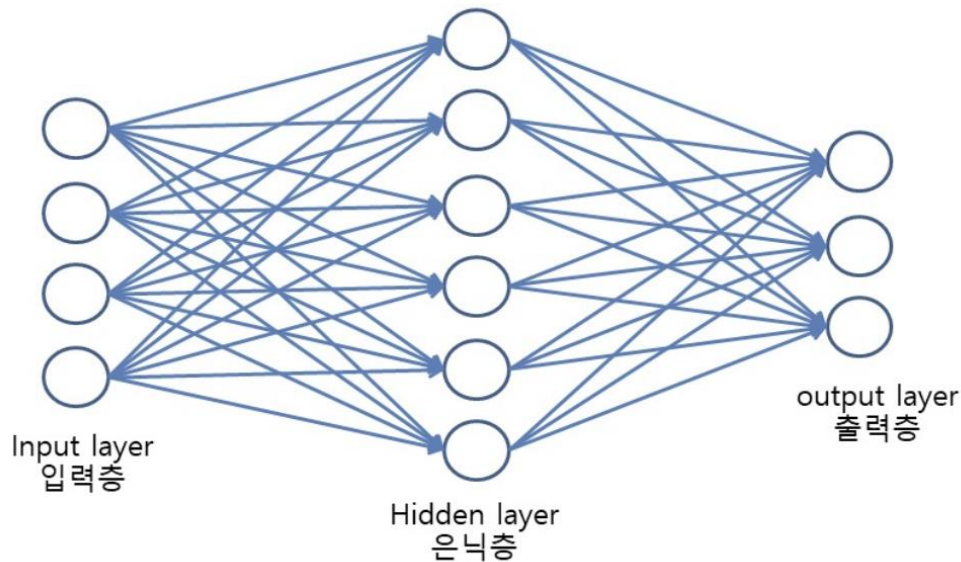
역전파



예측값과 실제값의 오차를 계산
오차가 각 층에 기여한 정도를 역순으로 전달, 가중치 업데이트 한다

3. 기본 개념 알고가기 입출력과 가중치?

입력과 출력? 은닉층?



Input_size : 입력층 노드 수

Hidden_size : 은닉층 노드 수

Output_size : 출력층 노드 수

Batch Size: 한번에 들어갈 데이터의 수

W : 가중치 : 노드 간 곱해지는 값

b : 편향 : 노드 간 더해지는 값

Epoch : 전체 학습을 한 번 학습하는 단위

저 노드에서 일어나는 연산 $y = wx + b$

제일 처음 w, b 는 랜덤으로 생성!

3. 기본 개념 알고가기 배치 사이즈

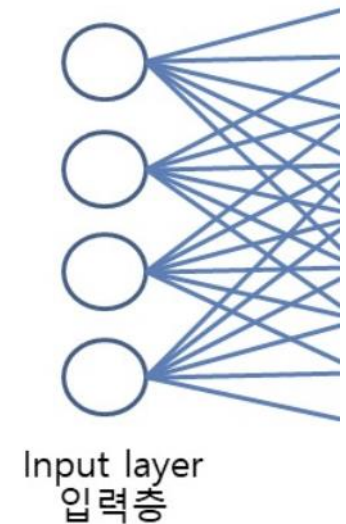
입력될 데이터의 크기를 정해보자 강아지 울음소리와 고양이 울음소리를 구분해 보자!

입력층 노드에 한 번에 얼마만큼의 데이터가 들어갈 지 정하는 값

미야옹 야옹 키야옹

고양이의 울음소리 예시 텍스트 32개

야옹야옹 응야냐 냐옹



데이터가 어떤 데이터인지에 따라 1 batch 가 의미하는 내용이 달라진다

1 batch in image: one image, in language: one token, in Geometry: one points

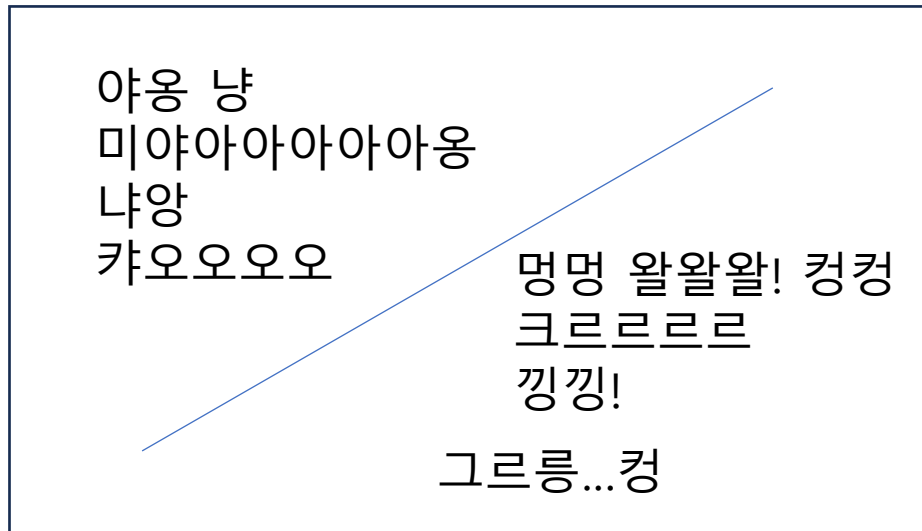
3. 기본 개념 알고가기 선형과 비선형

선형과 비선형

그런데, 고양이 울음소리에 **그르릉...컹!** 이 있다면??

강아지 울음 소리와 너무 비슷하잖아!

고양이 울음소리



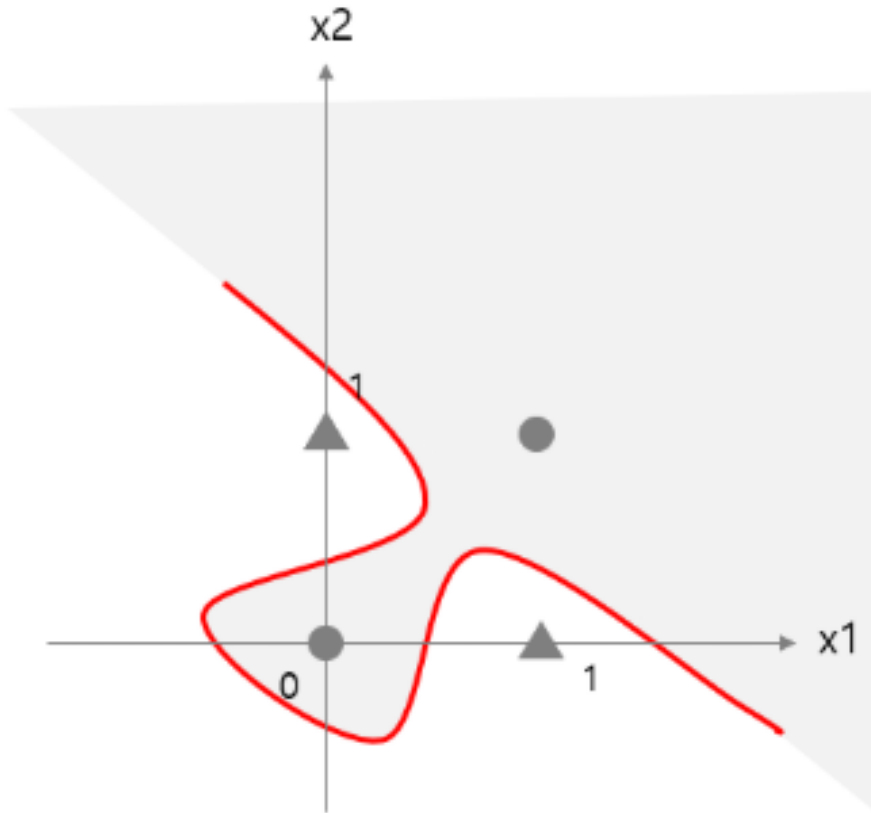
강아지 울음소리

$$y = wx + b$$

직선이기에, 절대 그르릉 킁을 하나의 선으로 구분할 수 없다!

3. 기본 개념 알고가기 선형과 비선형

선형과 비선형?



각 층의 노드 출력에 비선형성을 추가하는 함수

대표적으로, Sigmoid, Relu 등이 있다.

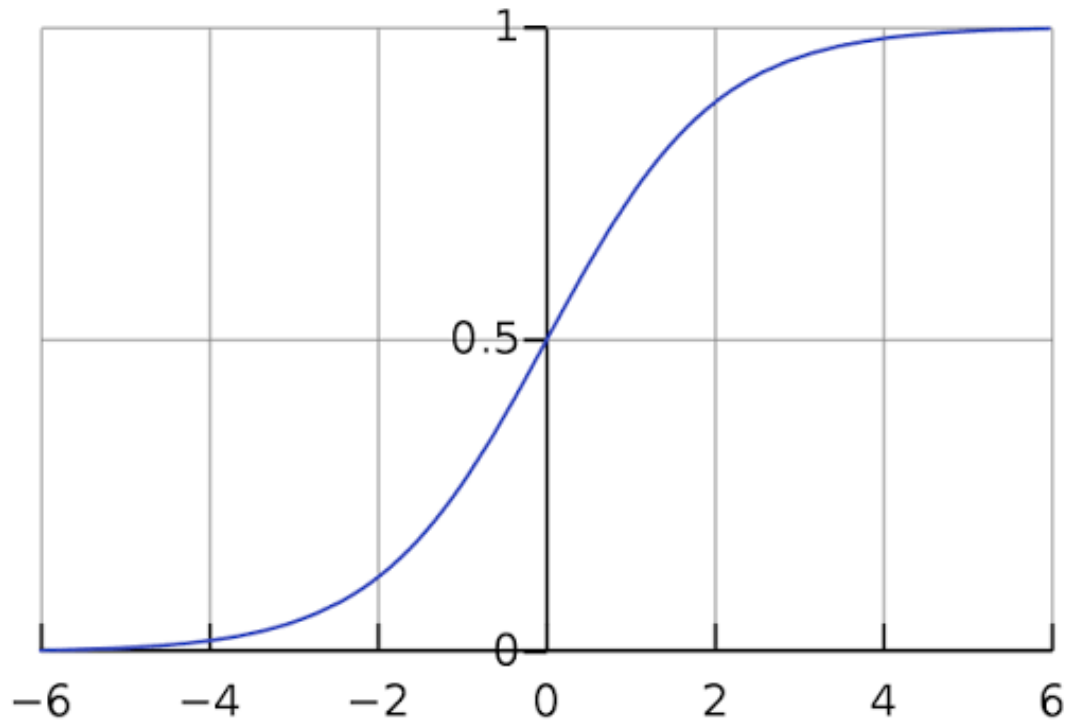
선형적인 딥러닝은 활성화 함수 없이 가능하다

비선형 적이면 활성화 함수가 들어야 한다

예시로, 활성화 함수 없으면 xor 문제를 학습할 수 없다.

3. 기본 개념 알고가기 활성화 함수 Sigmoid

Sigmoid



$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

노드의 출력값을 이 식에 집어넣는것으로 선형성을 부여한다!

왜 이렇게 복잡하게 생겼을까??

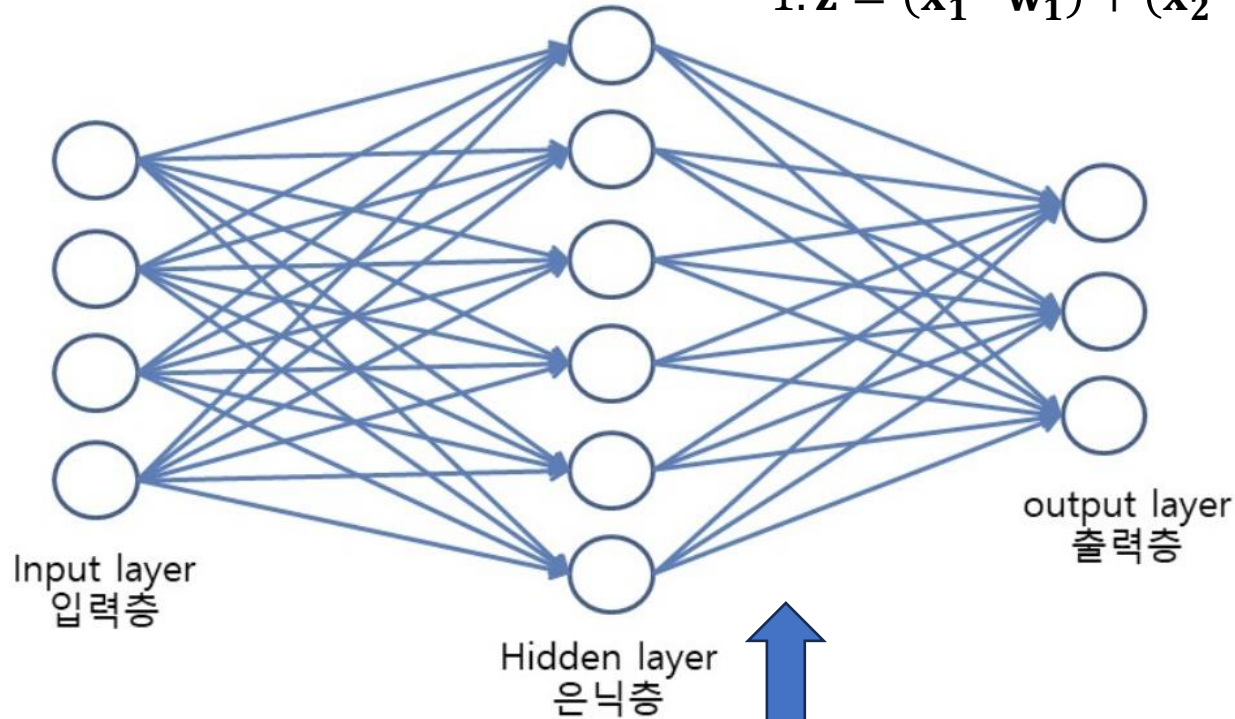
3. 기본 개념 알고가기

은닉층에서의 출력값

이 선 하나하나에 w 가 있다. (w_1, w_2, w_3, \dots)

$$1. z = (x_1 \cdot w_1) + (x_2 \cdot w_2) + (x_3 \cdot w_3) + (x_4 \cdot w_4)$$

입력값 x 



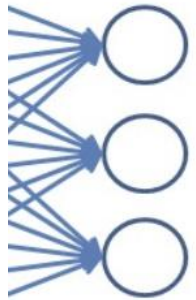
$$2. z' = z + b$$

$$3. y = \sigma(z')$$


$$y = \sigma(Wx + b)$$

3. 기본 개념 알고가기 출력층

출력층



output layer
출력층

$$1. z = (x_1 \cdot w_1) + (x_2 \cdot w_2) + (x_3 \cdot w_3)$$

$$2. z' = z + b$$

$$3. y = \sigma(z')$$

하지만, 여기서의 활성화 함수는 시그모이드가 아니다!

출력층의 활성화함수는 바로 소프트맥스!

3. 기본 개념 알고가기 출력층

출력층

$$y_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

생긴건 복잡하지만, 노드 3개의 합을 1로 만들어주는 역할이 전부!

- 2.0
- 1.0
- 0.1

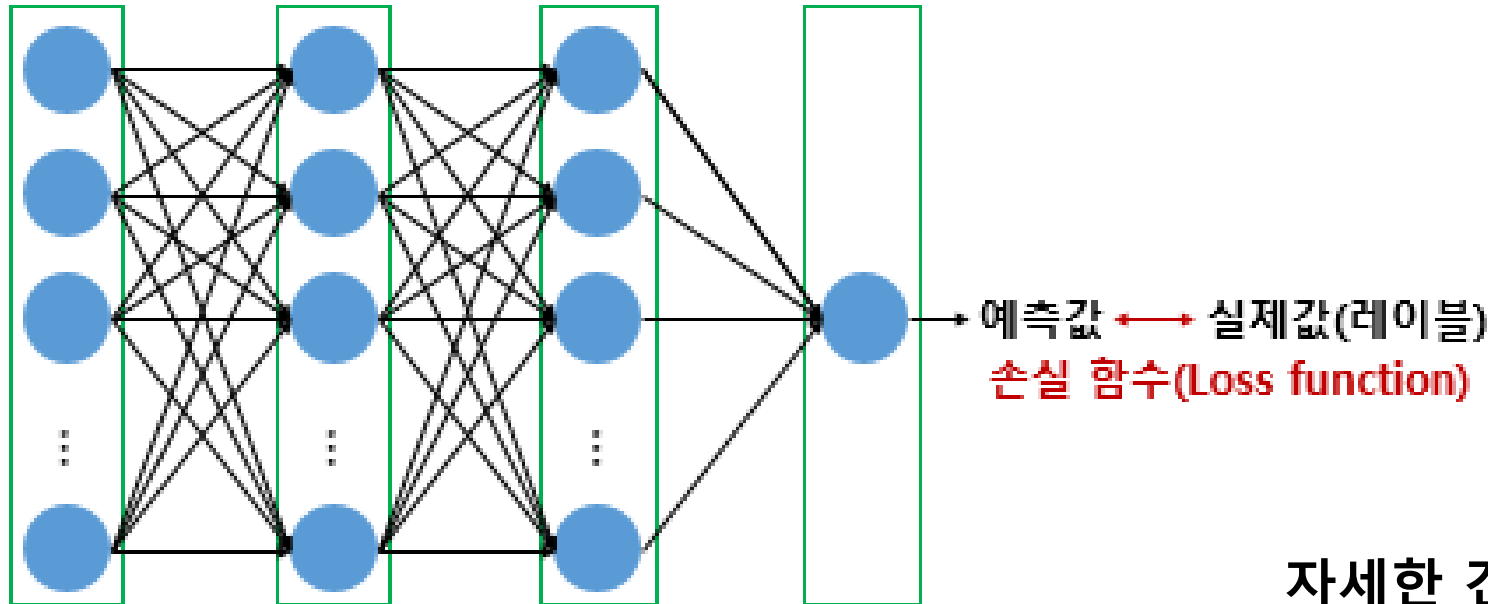


- 0.66
- 0.24
- 0.10

3. 기본 개념 알고가기 손실함수

✓ 오차를 최소화하는 방향으로 모델을 학습시키기 위함 이다

여기까지가 순전파!



모델의 예측값과 실제값 간의 차이(오차)를 측정하는 함수 MSE, MAE 등이 있다.

자세한 건 나중에 알아보자

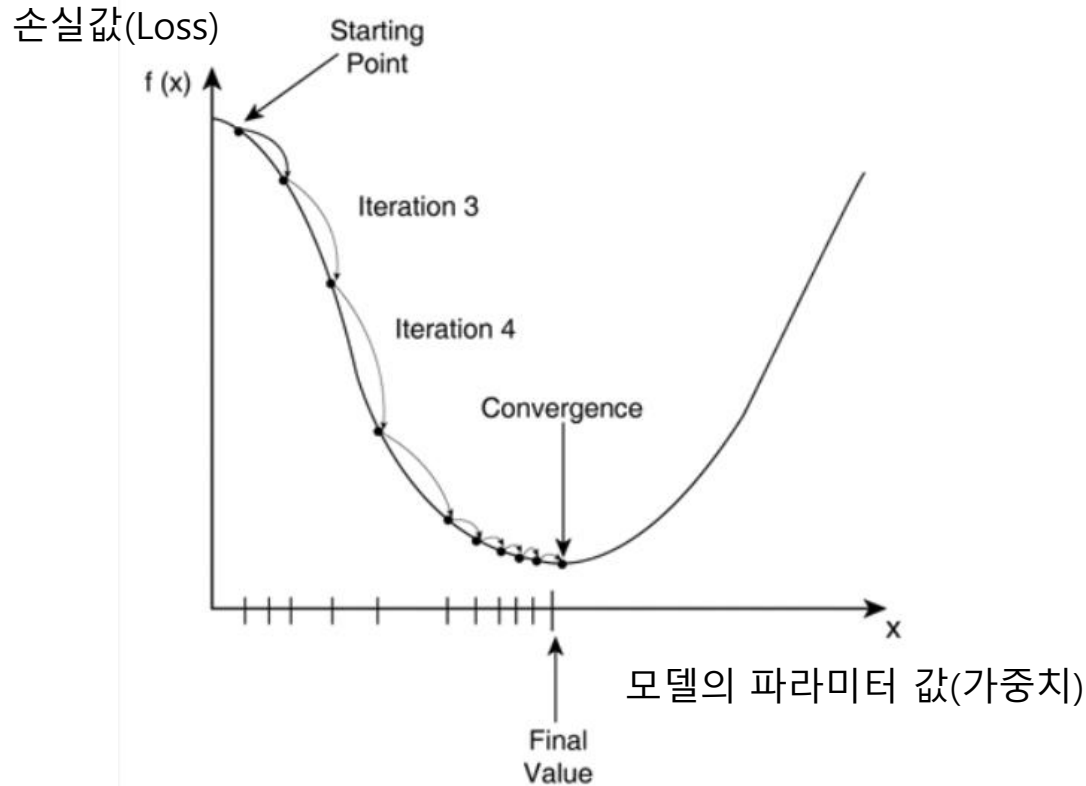
실제 정답과 모델의 예측값을 비교!
얼마나 틀렸는지 알아낸다.

https://gihak111.github.io/ai/2025/09/18/MSE_upload.html

https://gihak111.github.io/ai/2025/09/18/MAE_upload.html

3. 기본 개념 알고가기 경사하강법

여기서부터 역전파



오차를 최소화 하는 방향으로
가중치, 편향을 업데이트 한다

$$w_{\text{new}} = w_{\text{old}} - \eta \nabla J(w)$$

새로운 위치 = 현재 위치 + 움직인 방향과 거리

w_{new} : 업데이트 된 가중치. 다음 학습에서 사용된다.

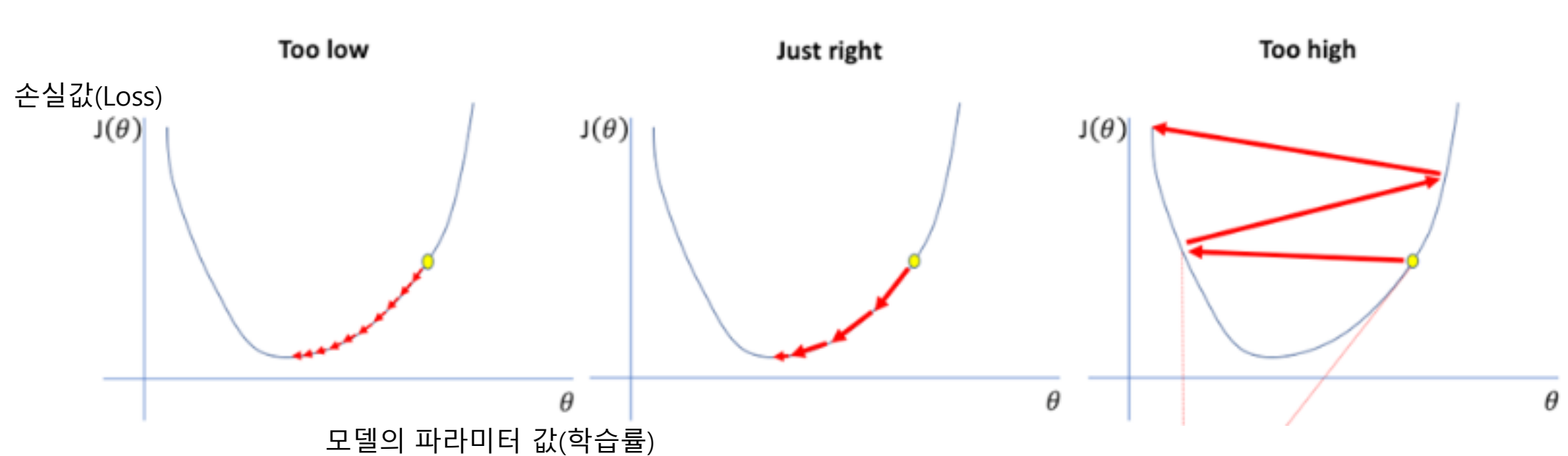
w_{old} : 현재 가중치. 출발하는 값.

η : 학습률. 경사를 얼마나 많이 이동할 지 정한다.

$\nabla J(w)$: 현재 위치에서 기울기가 가장 가파른 방향.

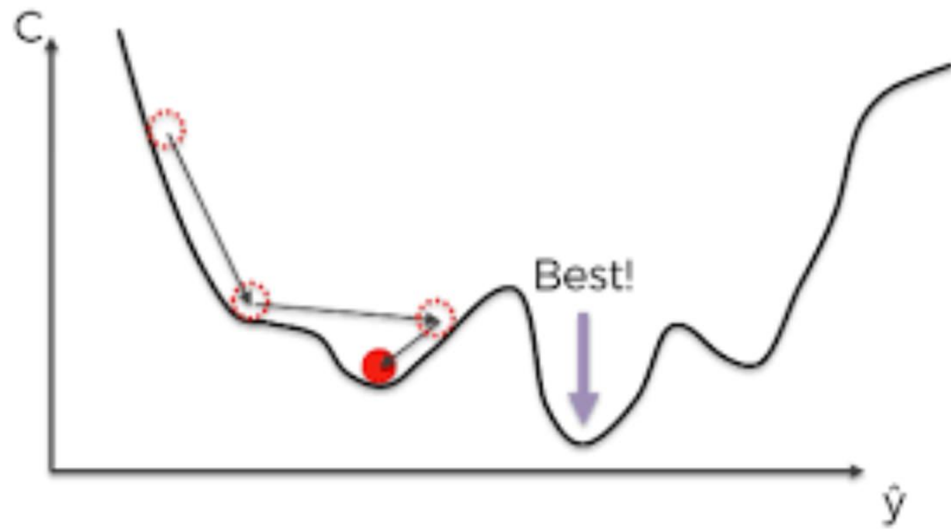
여기에 사용되는 기울기를 역전파가 알려준다

3. 기본 개념 알고가기 학습률



경사를 따라 얼마나 크게 이동할지 정하는 하이퍼파라미터 값.
적절한 값을 찾지 않으면 딥러닝이 망한다.

3. 기본 개념 알고가기 학습률과 경사하강법



학습률이 너무 작으면 지역 최솟점으로 수렴해 빠져 나오지 못한다
전역 최솟점으로 가기 위해 적절한 값을 설정해야 한다

1000 Epoch 면, 위와 같은 과정을 1000번 반복해서 더욱더 올바른 w , b 값을 찾아낸다.

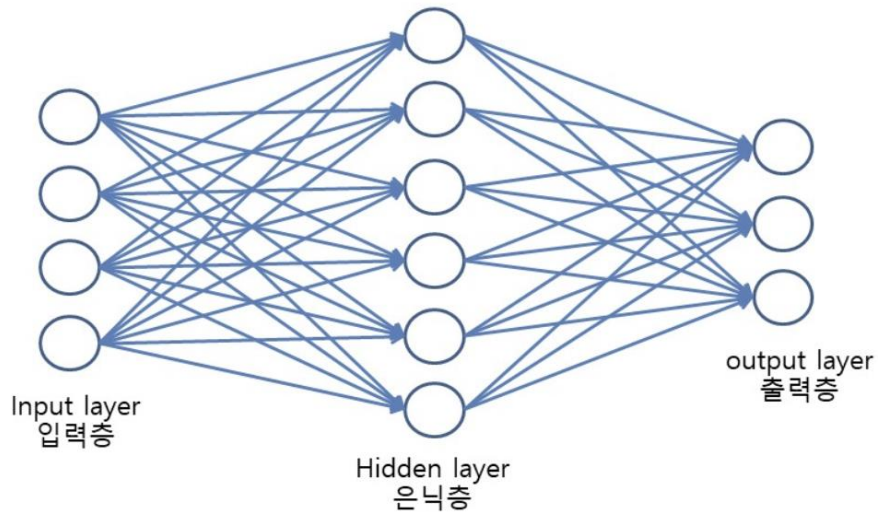
값을 찾아 업데이트 하는 과정이 역전파!

이걸 한 번 반복하는 것이 1 Epoch!

3. 기본 개념 알고가기 딥러닝 진행 단계



1. 순전파 : 예측값 계산



3. 가중치 및 편향 업데이트



2. 역전파 : 기울기 계산

3. 기본 개념 알고가기 딥러닝 진행 단계

ONECLICK AI

<https://oneclickai.co.kr/train/domain/regression/0>



4. Colab으로 MLP 알아보기 라이브러리?

ONECLICK AI

이렇듯, 반복을 통해 학습할 수 있는 AI!
위 내용을 직접 실행해 보려면 어떻게 해야 할까??



NumPy



PyTorch



TensorFlow

하지만, 설치도 어렵고 상황에 따라서 비싼 GPU를 필요로 하는 딥러닝! 어떻게 하면 더 간단히 할 수 있을까??

4. Colab으로 MLP 알아보기 라이브러리?

ONECLICK AI

Google Colab

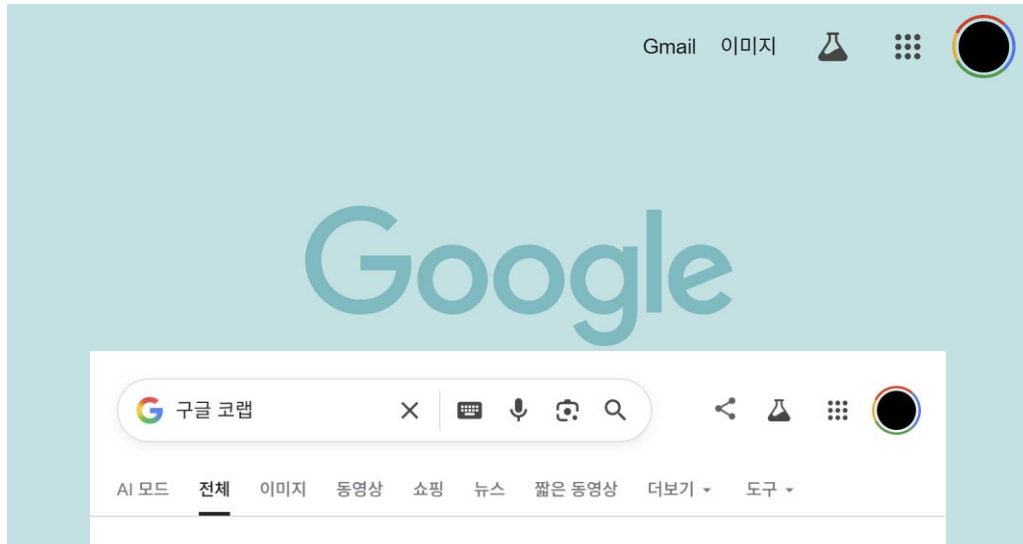
A100 GPU 를 공짜로 사용할 수 있다!



앞서 나온 파이썬이나 다른 라이브러리 설치 없이 바로 실행할 수 있다!

4. Colab으로 MLP 알아보기 라이브러리?

ONECLICK AI



노트 열기

예 >

노트북 검색

최근 사용 >

제목

마지막 연 시간 ↓ 처음 연 시간

Google Drive >

app.ipynb

6월 15일

6월 8일

GitHub >

app.ipynb

6월 15일

6월 15일

업로드 >

app.ipynb의 사본

6월 12일

6월 12일

이미지학습.ipynb

6월 8일

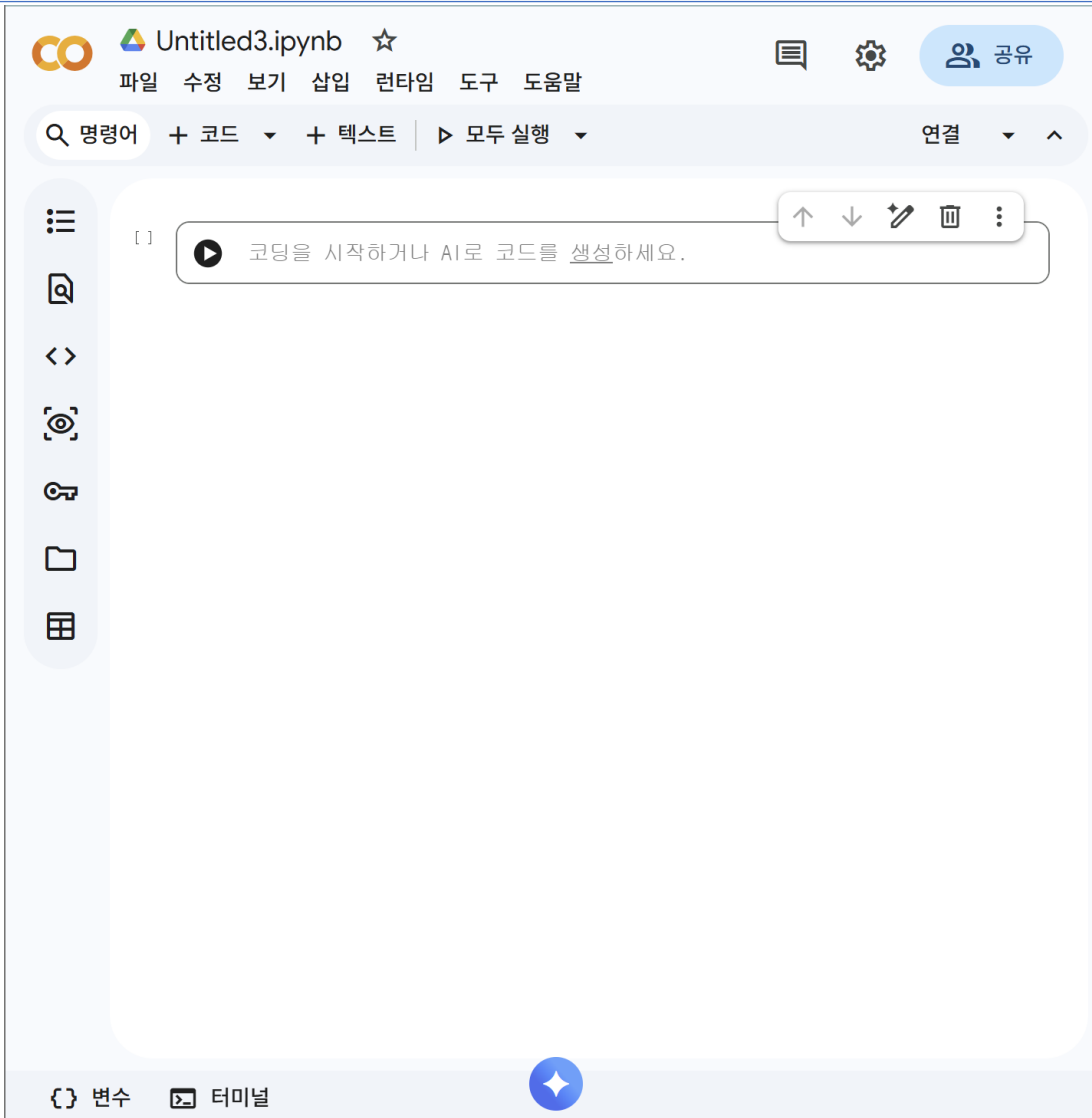
4월 20일

+ 새 노트

취소

4. Colab으로 MLP 알아보기 라이브러리?

ONECLICK AI



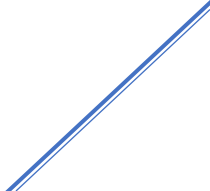
매번 치기 귀찮은 코드!
이를 해결해 주는 라이브러리!

10줄이 넘어가는 코드들을 한 줄로 줄여주는 `pytouch`

4. Colab으로 MLP 알아보기 라이브러리?

ONECLICK AI

https://gihak111.github.io/ai/2026/06/08/AI_Learn_with_Colab_upload.html



4. Colab으로 MLP 알아보기 라이브러리?

ONECLICK AI

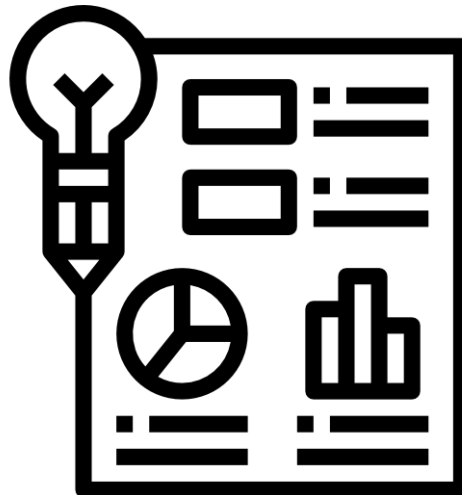
MLP는 가장 기본적인 형태의 신경망으로,
제미나이나 챗 지피티, 클로드코드의 오픈스 등 세상에 존재하는 모든 AI에 반드시 들어가는 형태

5. 정보통신과 AI 딥러닝이 진행되는 단계?

ONECLICK AI

정보통신에서 다루는 데이터에는 어떤게 있을까?

AI가 다룰 수 있는 데이터?

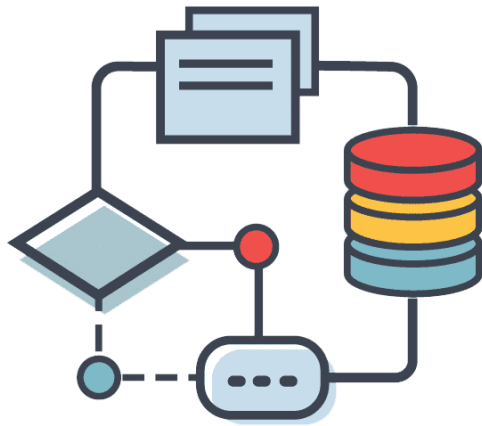


5. 정보통신과 AI 딥러닝이 진행되는 단계?

알고리즘과 딥러닝의 장단점?

식을 명확하게 알고 있으면 알고리즘이 우세!

식을 명확하게 모르고 있다면 딥러닝이 우세!



Algorithm



5. 정보통신과 AI 딥러닝이 진행되는 단계?

ONECLICK AI

어? 그러면 정보통신에서 AI 를 이용한다는 것 자체가 기존 방식인 알고리즘에 비해 비효율 적인거 아닌가요?

맞다. 하지만, 경우에 따라선 AI를 사용하는게 유리할 때도 있다
이 경우에 대해서 다음 시간에 알아보도록 하자

딥러닝과 머신러닝의 차이
딥러닝에 필요한 개념 정리
MLP 알아보기
정보통신과 AI

감사합니다