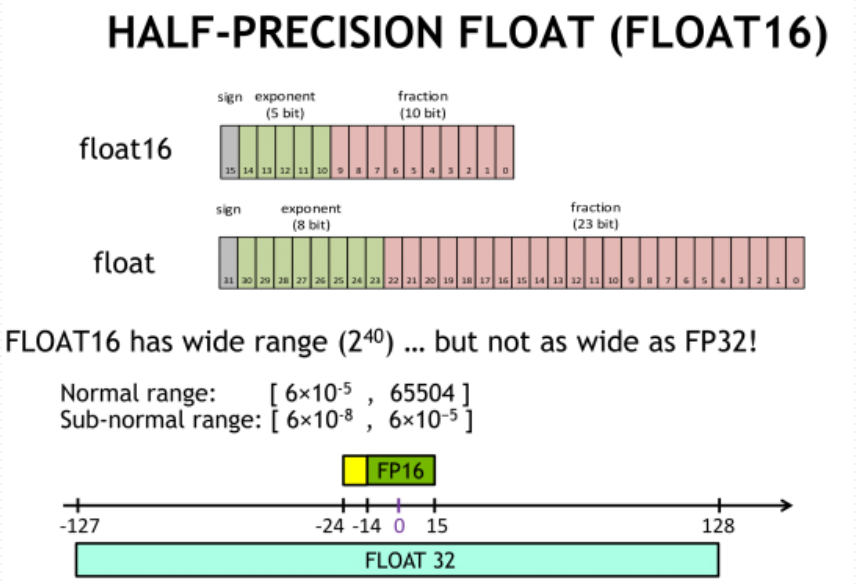
AI 용어정리 - Data type & precision of AI

AI 모델에서 INT8/FP16/FP32 와 같은 용어들이 나오네. 이게 뭐지?

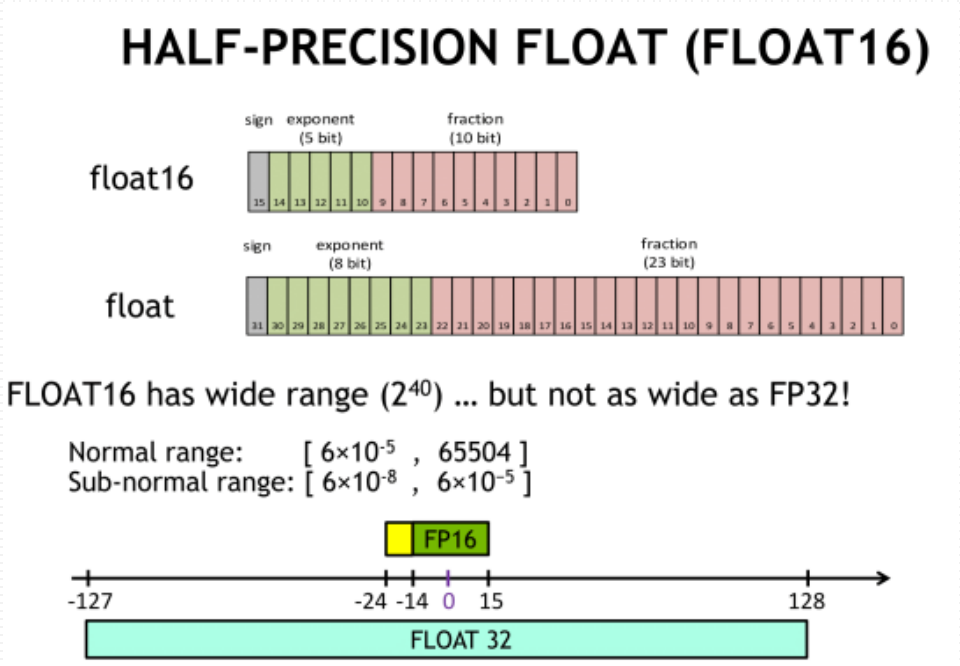
- INT8, FP16, FP32 등의 표현은 AI 모델의 가중치나 연산에서 사용되는 데이터의 형식과 정밀도를 의미
- 데이터의 형식과 정밀도는 모델의 학습과 추론에 큰 영향을 미치며, 높은 정밀도는 더 정확한 연산을 제공하지만, 더 많은 메모리와 연산 리소스를 필요로 함. 반면에 낮은 정밀도는 더 빠른 연산과 메모리 절약을 가능하게 하지만, 모델의 정확도가 줄어 들 수 있음



1. **INT8 (8-bit integer)**
   - 정수 값을 8비트로 표현.
   - 양자화된 모델에서 자주 사용되며, 경량화와 추론 속도 향상을 위해 사용
2. **INT16 (16-bit integer)**
   - 정수 값을 16비트로 표현
   - 일부 중간 정밀도 연산이나 특정 하드웨어에서의 연산을 위해 사용
3. **INT32 (32-bit integer)**
   - 정수 값을 32비트로 표현
   - 일반적인 CPU 연산 및 일부 고도의 정밀성이 필요한 연산에 사용
4. **FP16 (16-bit floating point) or Half precision**
   - 숫자를 16비트 부동소수점으로 표현
   - GPU 연산 최적화와 모델의 경량화를 위해 사용
5. **FP32 (32-bit floating point) or Single precision**
   - 숫자를 32비트 부동소수점으로 표현
   - 일반적인 GPU 연산 및 대부분의 훈련 연산에 사용
6. **FP64 (64-bit floating point) or Double precision**
   - 숫자를 64비트 부동소수점으로 표현
   - 고도의 정밀성이 필요한 연산 및 과학적 연구에서 사용
7. **BF16 (Brain Floating Point 16)**
   - 16비트 부동소수점이지만, FP16과 다른 비트 구성을 가짐
   - Google의 TPU와 같은 특정 하드웨어에서 사용
8. **TF32 (TensorFloat-32)**
   - NVIDIA의 최신 GPU에서 사용되는 특수한 데이터 타입
   - FP32의 정밀도와 FP16의 성능 사이의 중간 지점을 제공하기 위해 설계됨

AI 용어정리 - Data type & precision of AI

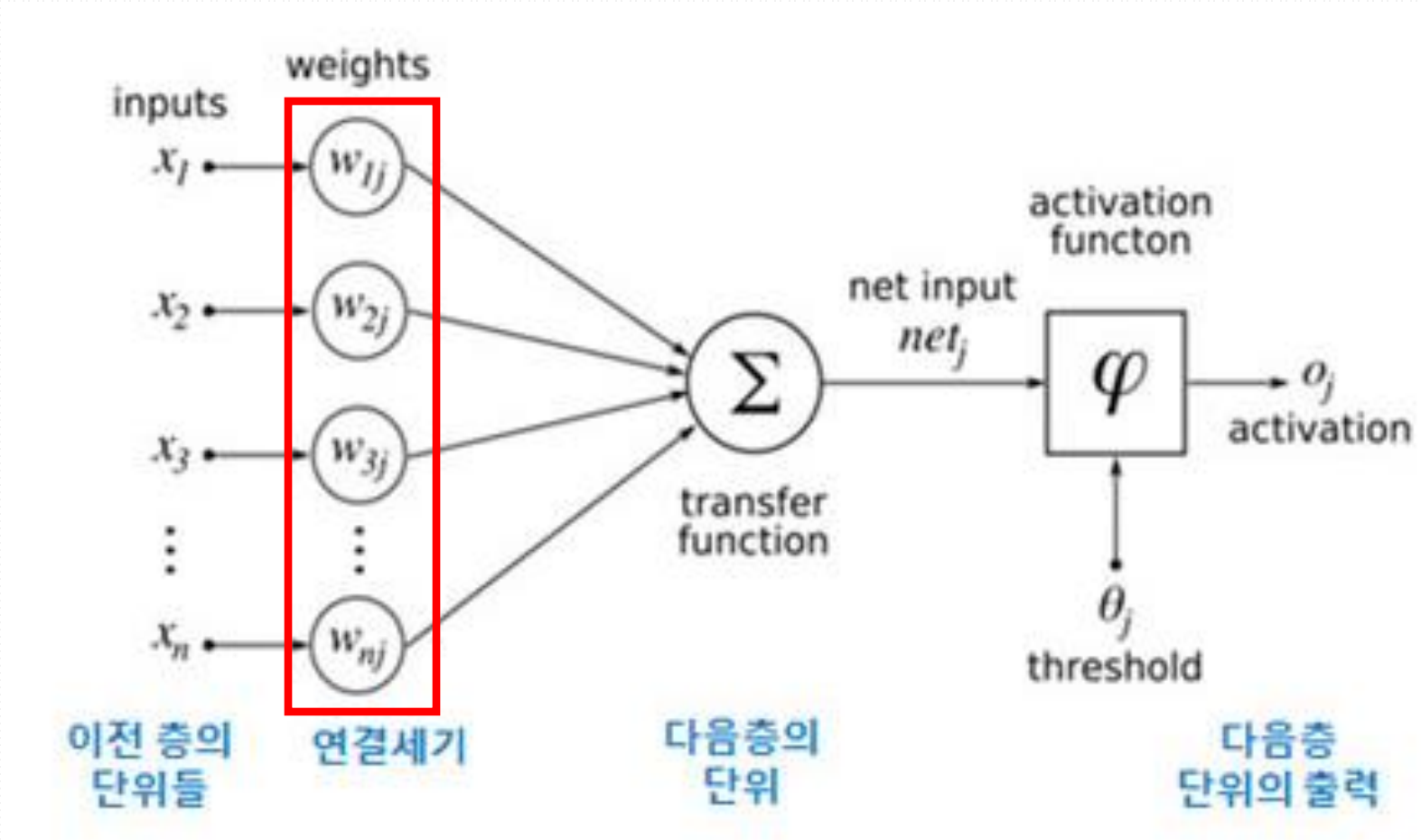
AI 모델에서 INT8/FP16/FP32 와 같은 용어들이 나오네. 이게 뭐지?



IEEE 754 표준

Half Precision (Floating Point 16)	**1 bit**	**5 bit**	**10 bit**
Single Precision (Floating Point 32)	**1 bit**	**8 bit**	**23 bit**
Double Precision (Floating Point 64)	**1 bit**	**11 bit**	**52 bit**
Quadruple Precision (Floating Point 128)	**1 bit**	**15 bit**	**113 bit**

AI 용어정리 - Data type & precision of AI



AI 용어정리 - GFLOP

Giga Floating Point Operation

Amount of single precision giga-floating point operations calculated

Most of deep learning calculation is proceed in floating point

1 Giga FLOPS (GFLOPS) computer system is capable of performing **one billion floating-point operations per second**

Easy to represent a large dynamic range

Mostly used in picture information (graphics)

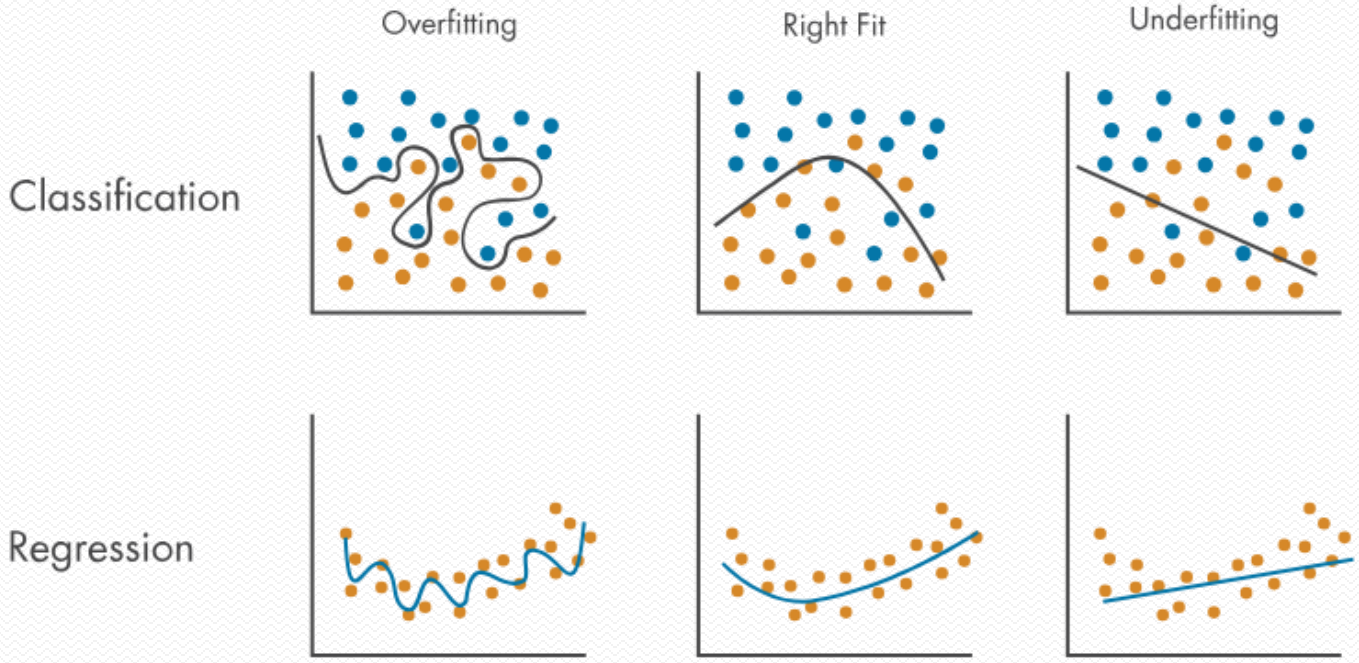
$$\text{FLOPS} = \text{cores} \times \text{clock} \times \frac{\text{FLOPs}}{\text{cycle}}$$

Unit	Flops
kFLOPS	10^3
MFLOPS	10^6
GFLOPS	10^9
TFLOPS	10^{12}
PFLOPS	10^{15}
EFLOPS	10^{18}
ZFLOPS	10^{21}

AI 용어정리 - Overfitting

Overfitting 의 개념은 뭐고 왜 발생하나?

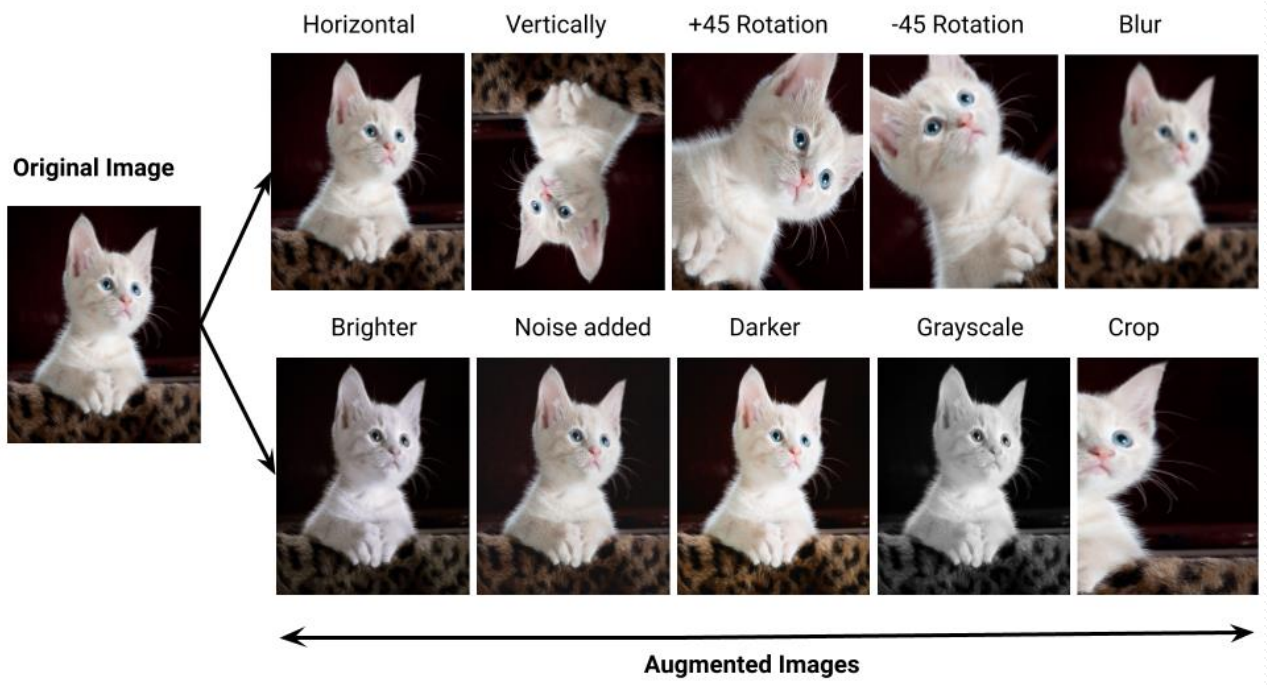
- 오버피팅(Overfitting)은 기계 학습 및 딥러닝에서 모델이 학습 데이터에 지나치게 최적화되어, 새로운 데이터에 대한 예측 성능이 떨어지는 현상을 의미
- 모델이 학습 데이터의 특정 노이즈나 상세한 패턴까지 너무 정확하게 학습하여 일반화 성능이 떨어진 상태
- 모델의 복잡도가 너무 높거나, 학습 데이터의 양이 부족할 때 주로 발생



AI 용어정리 - Data Augmentation

Data Augmentation 은 왜 필요하고 어떻게 가능할까?

- 데이터 증강(Data Augmentation)은 기존의 데이터셋을 활용하여 새로운 데이터를 생성하는 기법
- 주로 딥러닝이나 기계 학습에서 데이터의 양을 늘리기 위해 사용
- 데이터 증강은 원본 데이터에 약간의 변화(회전, 확대/축소, 반전 등)를 가하여 다양한 형태의 새로운 데이터를 만들어내는 방식
- 이를 통해 모델이 오버피팅(과적합)을 줄이고 일반화 성능을 향상시킴



AI 용어정리 - Transfer Learning

Transfer Learning 의 개념을 알아보고 활용해 보자.

- Transfer Learning은 사전에 훈련된 모델을 사용하여 새로운 작업을 더 빠르고 효과적으로 학습하는 기계 학습의 방법

- 기본 아이디어는 이미 큰 데이터셋으로 학습된 모델의 지식을 다른 관련된 작업에 전이하여 사용하는 것

- 실제 사용사례: 이미지 인식 - 이미지 분류를 위해 사전에 훈련된 모델이 있습니다. 이 모델은 수백만 개의 이미지를 사용하여 다양한 물체를 인식하는 방법을 배웠습니다. 이제 특정한 작업, 예를 들어 의료 영상에서 종양을 탐지하는 작업을 수행하려고 합니다. 종양 탐지를 위한 데이터는 제한적이지만, 사전에 훈련된 이미지 분류 모델의 지식을 활용하여 종양 탐지 모델을 더 빠르게 학습시킬 수 있습니다. 이미지 분류 모델에서의 일반적인 이미지 인식 능력이 종양 탐지 작업에 도움을 주는 것

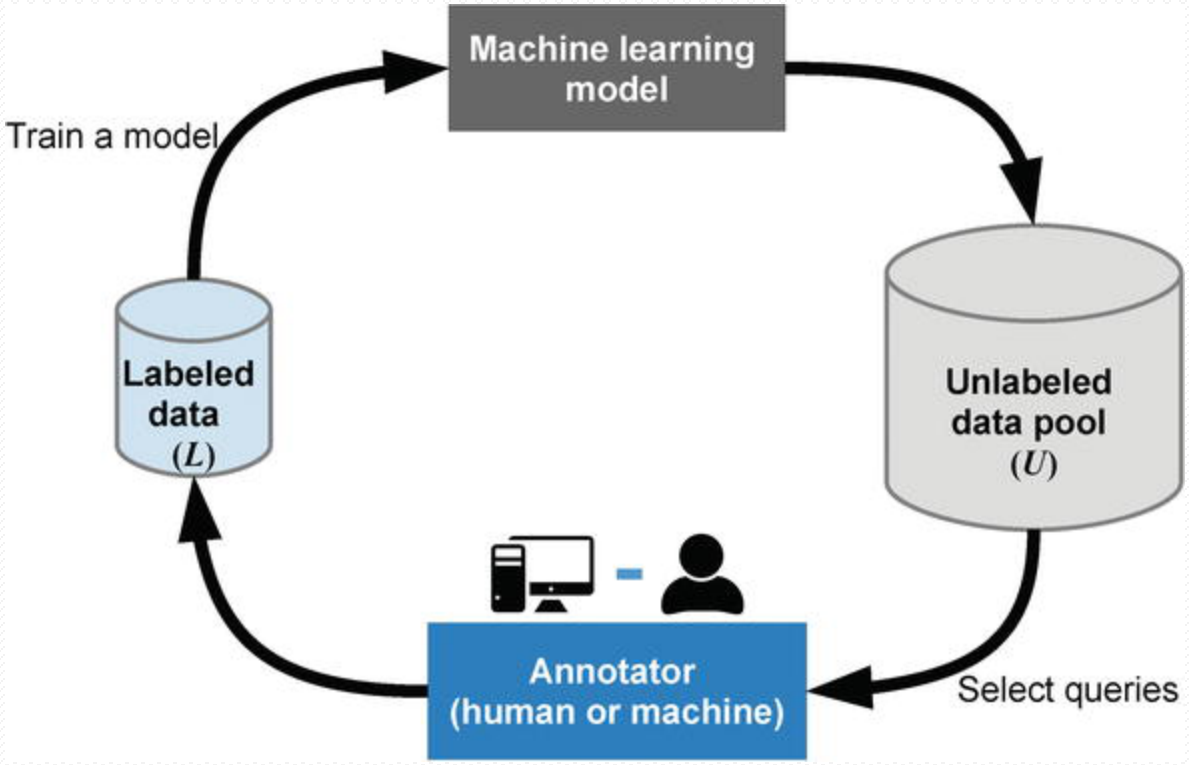
```python
import tensorflow as tf
from tensorflow.keras.applications import VGG16
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Flatten
from tensorflow.keras.datasets import cifar10

# 데이터 불러오기 및 전처리
(train_images, train_labels), (test_images, test_labels) = cifar10.load_data()
train_images = train_images / 255.0
test_images = test_images / 255.0

# 사전 훈련된 VGG16 모델 불러오기
base_model = VGG16(weights='imagenet', include_top=False, input_shape=(32, 32, 3))

# 모델의 가중치를 고정 (학습되지 않도록 설정)
for layer in base_model.layers:
    layer.trainable = False

# 사용자 정의 분류기 추가
model = Sequential([
    base_model,
    Flatten(),
    Dense(512, activation='relu'),
    Dense(10, activation='softmax')  # CIFAR-10은 10개의 클래스를 가짐
])

# 컴파일 및 학습
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['a
model.fit(train_images, train_labels, epochs=5, batch_size=64, validation_data=(tes
```

AI 용어정리 - Active Learning

Active Learning 의 개념과 장점을 알아보자.

- 모델 스스로 학습에 가장 효과적인 데이터 샘플을 선택하여 라벨링을 요청하는 방식을 의미

- Example) 상상해보세요. 당신은 수천장의 사진을 가지고 있고, 이 사진들 중에서 고양이 사진만을 분류하려고 합니다. 그런데, 각 사진에 라벨을 붙이는 것은 시간과 비용이 많이 듭니다. 여기서 Active Learning을 사용하면, 처음에는 몇 장의 사진만 라벨링하고 이를 기반으로 초기 모델을 학습시킵니다. 그 다음, 모델은 자신이 확신이 가지 않는 사진들 중에서 가장 '의심스러운' 사진 몇 장을 선택하여 사용자에게 라벨링을 요청합니다. 이렇게 선택적으로 라벨링을 진행하면서, 모델은 점점 더 정확해지게 됩니다. 결과적으로, 모든 사진을 라벨링하는 것보다 훨씬 적은 수의 사진만 라벨링하면서도 높은 정확도의 모델을 얻을 수 있습니다.



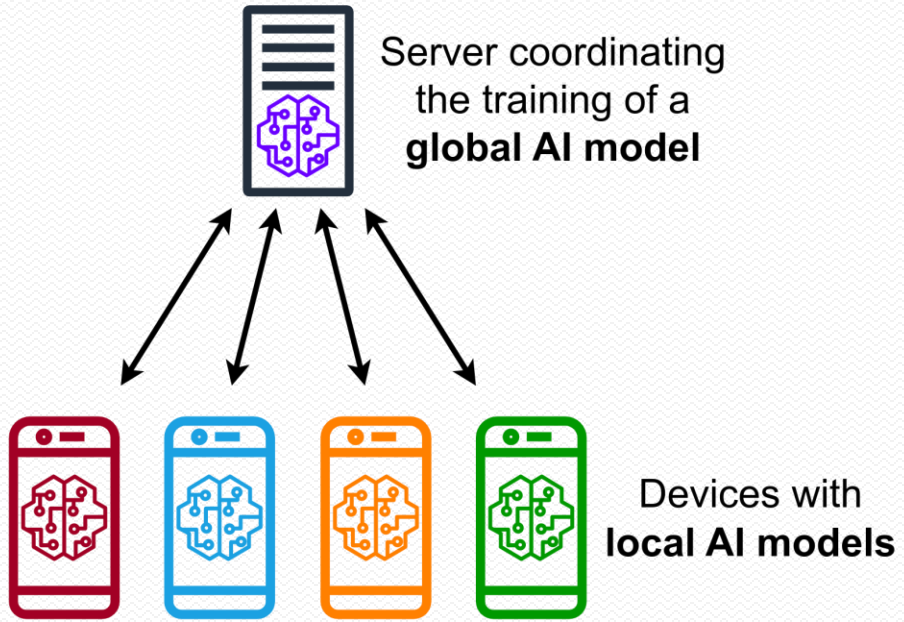
AI 용어정리 - Federated Learning

Federated Learning 의 개념과 사용법

- Federated Learning은 분산된 여러 기기에서 모델 학습을 수행하고, 그 결과를 중앙 서버에서 집계하여 모델을 업데이트하는 방식의 기계 학습

- 핵심 원리
  1. **로컬 학습:** 각 기기는 자신의 데이터로 모델을 독립적으로 학습함
  2. **모델 업데이트 공유:** 각 기기는 로컬에서 학습한 모델의 업데이트를 중앙 서버로 전송
  3. **집계 및 통합:** 중앙 서버는 모든 기기로부터 받은 모델 업데이트를 통합하여 전체 모델을 업데이트
  4. **업데이트된 모델 배포:** 중앙 서버는 업데이트된 모델을 모든 기기에 배포

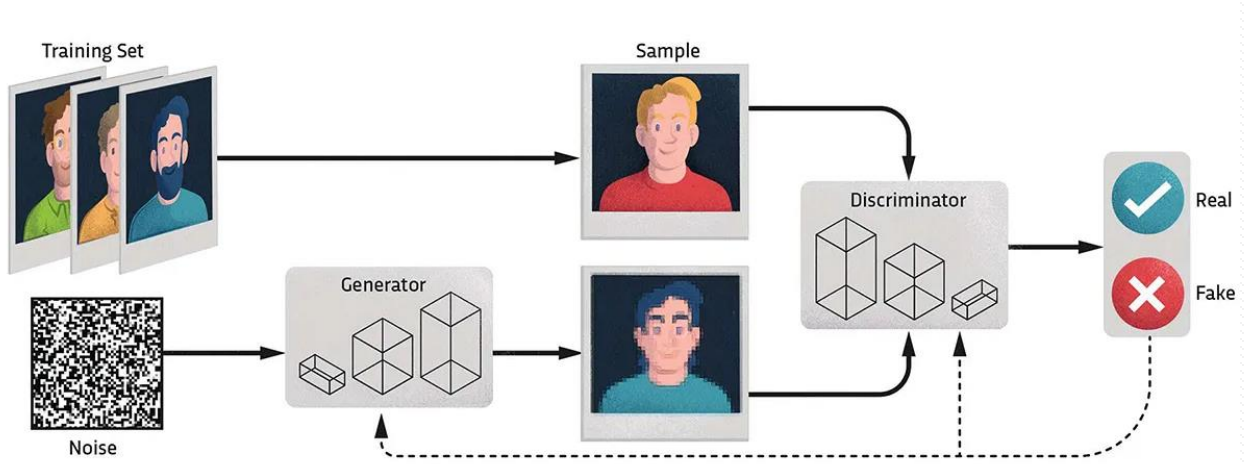
- 장점
  1. **데이터 프라이버시:** 데이터는 로컬 기기에 머무르므로 중앙 서버에 개인 데이터를 전송할 필요가 없어 의료, 금융 등 민감정보 포함하는 곳에 주로 사용
  2. **효율적인 대역폭 사용:** 모델의 파라미터만 전송되므로 대량의 데이터 전송이 필요 없음
  3. **실시간 학습:** 각 기기에서 실시간으로 데이터를 수집하고 학습할 수 있음

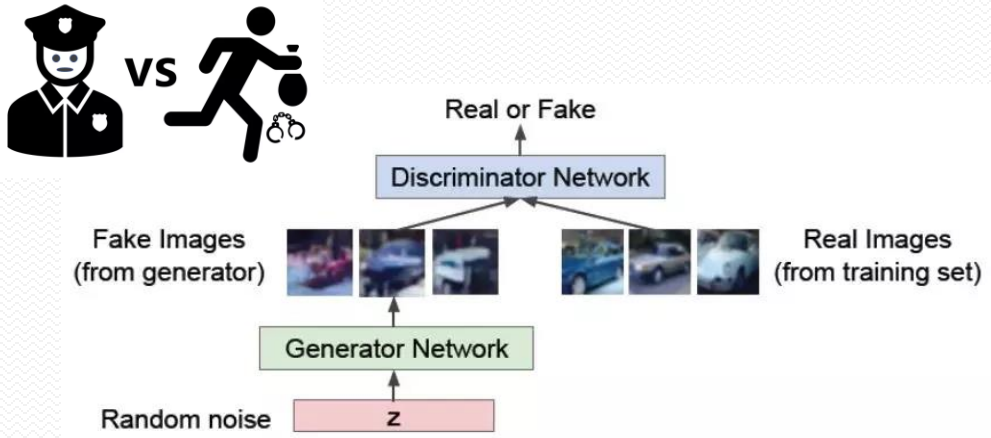


AI 용어정리 - Generative AI

생성형 AI 는 기존의 AI 모델들과 어떤 차이가 있을까?

- Generative AI는 데이터를 기반으로 새로운 내용을 생성할 수 있는 인공지능의 한 분야
- 이미지, 음악, 텍스트 등의 새로운 콘텐츠를 생성하는 것이 포함되며 대표적으로 Generative Adversarial Networks (적대적 생성 신경망, GANs)가 있음
- 기존의 AI는 주로 데이터를 분류하거나 예측하는 데 중점을 둔다면, Generative AI는 새로운 콘텐츠를 생성하는 것에 중점을 둠
- 훈련 방식의 차이
  - 기존의 AI: 주로 지도 학습 방식을 사용하며, 입력 데이터와 그에 해당하는 라벨을 사용하여 모델을 훈련
  - Generative AI: 주로 비지도 학습 또는 준지도 학습 방식을 사용하며, 데이터의 분포를 학습합니다. GAN의 경우, 생성자와 판별자 두 개의 네트워크가 경쟁하는 방식으로 훈련





AI 용어정리 - LLM (Large Language Model)

LLM 이라고 자꾸 사람들이 말한다. 대체 기존의 AI 와 뭐가 다른가?

- **NLP** (자연어 처리, Natural Language Processing):
  - 정의: NLP는 컴퓨터가 인간의 언어를 이해하고 처리할 수 있도록 도와주는 컴퓨터 과학 및 인공 지능의 하위 분야
  - 목적: 텍스트나 음성 데이터를 분석, 이해 및 생성하기 위한 기술들을 개발하는 것
  - 예시: 기계 번역, 감정 분석, 텍스트 요약, 음성 인식 등이 NLP의 응용 분야임

- **LLM** (대규모 언어 모델, Large Language Model):
  - 정의: LLM은 방대한 양의 텍스트 데이터를 학습하여 자연어 처리 작업을 수행하는 딥러닝 모델
  - 목적: 복잡한 언어 패턴을 학습하여 다양한 NLP 작업에서 높은 성능을 달성하는 것
  - 예시: OpenAI의 GPT (Generative Pre-trained Transformer) 시리즈와 같은 모델들이 LLM의 대표적인 예

- LLM이 NLP 분야의 일부분임