

딥러닝

📅 날짜	@2024년 9월 3일
☰ 태그	딥러닝

학습과정에서 학습계층이 있다.

- 입력층, 출력층 사이 → 은닉층이 몇겹인지가 딥러닝의 핵심이다.
- 입력층 데이터 → 은닉층

딥러닝

1. 알고리즘 개발
2. 있는 모델을 활용을 한다.

그리드서치 적극적 활용

딥러닝은 항상 지도 학습

분류 또는 예측 모델

1. 처리
2. 모델 생성

일부 데이터는 머신러닝이 더 정확할 수 있다. 또는 머신러닝보다 그냥 하는게 더 정확할 수 있다.

데이터에 따라서 모델 선택을 해야한다.

데이터 적음 → 많음

단순 연산 → 머신러닝, 딥러닝

데이터의 복잡도가 단순하면 통계분석 → 높으면 높을수록 딥러닝

통계모델/머신러닝/딥러닝 비교표

	통계모델	머신러닝	딥러닝	비고
데이터 복잡도	낮음	보통	높음	항상 비례하지 않음

	통계모델	머신러닝	딥러닝	비고
데이터의 양	적음	보통	많음	항상 비례하지 않음
처리시간	짧음	보통	김	
연산능력 요구	적음	보통	많음	
과적합 등의 위험	없음	보통	많음	

CPU 연산 - 모든 연산

- 정밀 연산

GPU 연산 - 특정 부분 연산

- 연산을 위해 정밀도 희생
- 부동소수점 : 특정 소수 숫자가 있을 때
- LLM : GPU

<https://hongong.hanbit.co.kr/딥러닝-프레임워크-비교-텐서플로-케라스-파이토치/tensorflow>

픽셀-rgb-8bit(255,255,255)

딥러닝은 복잡도에서는 $5 \times 5 \times 3$ 75차원이다

5×5 이미지에 rgb가 있다.

min-max scale 을 해준다.

이미지의 경우는 255로 나눈다.

tensorflow 딥러닝 모델 만드는법

1. 모델 정의
 - a. 어떤 형태의 모델을 구현할 것인가, Sequential model
2. 입력층 설정
 - a. Input shape, 어떤 형태의 데이터를 입력받을 것인가
 - b. 형태 변환
3. 출력층 설정

- a. binary/multiclass classification 선택
 - b. binary : 출력 1개 (0에 가깝냐 1에 가깝냐)
 - c. multiclass : 차원 개수 만큼 출력 설정
4. 은닉층 설정
- a. 데이터의 차원 또는 정보를 변형해가면 변환할 수 있도록 설계
5. 모델 컴파일
- a. 최적화 함수, 손실 함수, 평가 지표 등을 설정
6. 예측
7. 검증

예시

```
model = tf.keras.models.Sequential(
    [
        tf.keras.layers.Flatten(input_shape=(28, 28)), # 입력
        tf.keras.layers.Dense(128, activation='relu'), # 입력
        tf.keras.layers.Dropout(0.2),
        tf.keras.layers.Dense(10, activation='softmax') # 출력
    ]
)
```

Flatten Layer : 1차원으로 펼쳐주는 레이어.

Dense Layer : 입력 레이어, 출력 레이어를 연결해주는 레이어

- activation : 활성화함수

Dropout Layer : 은닉층의 노드를 랜덤으로 꺼버린다. (cross validation 같은)

- 과적합을 막기 위해서 사용

```
model.compile(
    optimizer='adam',
    loss='',
    metrics=['']
)
```

optimizer : 최적화 함수

loss : 오차 계산을 어떻게 할 것인가

딥러닝 활용

- 얼굴인식
- 생성형 이미지
- 소라, 디퓨전 ai, 디퓨저 비디오 같은 것
- 아이폰 얼굴인식(미모지 같은) face mesh, mediapipe
-

생성형 AI

- 데이터가 부족해서 만들어서 사용
-

zero-shot

q-shot-learning