

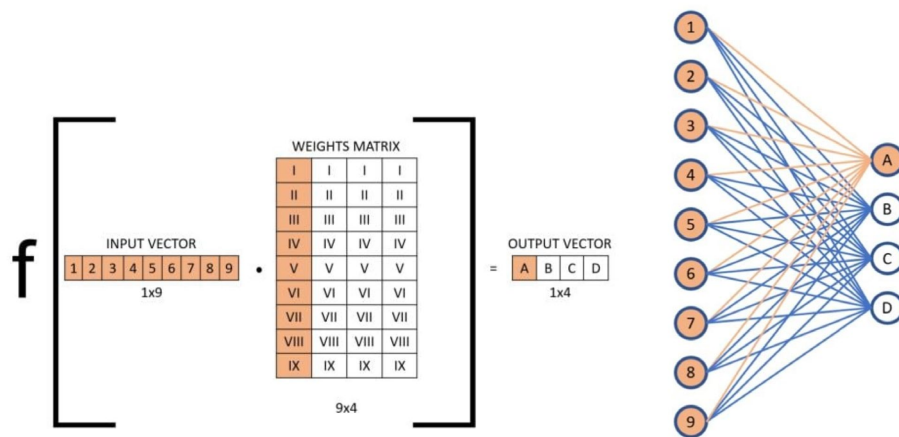
가) CNN(Convolutional Neural Networks)

1) CNN 정의

- 주로 영상이나 이미지 데이터를 처리할 때 사용
- 공간적/지역적 정보들이 손실되는 문제점을 해결하기 위해 나온 해결책
- 모델에 필요한 이미지 부분을 잘라서 학습 후 예측한다.

2) 완전 연결 계층(fully connected layer)

- 각 뉴런이 가중치(필터) 행렬을 통해 입력 벡터에 선형 변환을 적용하는 신경망
- 모든 입력 벡터가 모든 출력 벡터에 영향을 미침
- 이미지를 분류하고 설명하는 데 적합

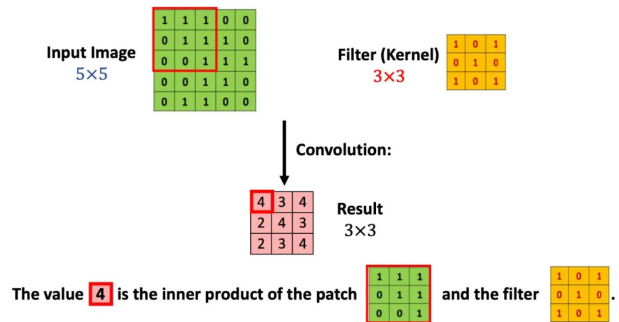
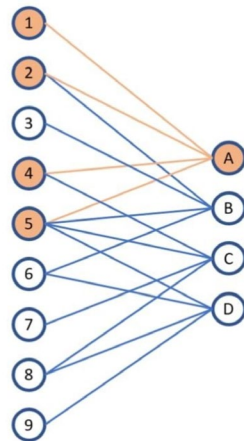


● 문제점

- Dense로 인해 용량이 증가한다.
- 과적합이 생길 가능성이 크다
- 이미지나 기타 구조화된 데이터의 공간적 계층을 활용하지 않아 사용하지 않는 공간이 생긴다 즉, 공간 이미지 학습 효과가 떨어질 수 있다. (3차원 데이터 -> 1차원 데이터)

3) 합성곱 계층(convolutional layer)

- 모든 입력이 모든 출력에 영향을 주지 않는다
- 완전 연결 계층의 문제점을 해결 (3차원 데이터 -> 3차원 데이터)



- 문제점

- 크기가 작아지면서 어느 시점에 합성곱 연산을 할 수 없게 된다.

4) 패딩(Padding)

- Conv 연산에서 생성된 출력이 레이어마다 줄어드는 것을 해결하기 위해 사용
- Edge 픽셀에 적절하게 값을 넣어 손실될 수 있는 정보들을 보호
- 출력 피쳐 맵의 크기를 제어할 수 있다.

- 패딩의 종류

(1) 유효 패딩(valid padding)

- 패딩을 포함하지 않는다.

(2) 동일 패딩(same padding)

- 출력 차원이 입력 차원과 동일하도록 패딩을 입력에 추가한다. 일반적으로 0을 넣은 픽셀을 추가

5) 풀링층(Pooling layer)

- 합성곱층(CONV & ReLU) 이후 연산층
- 특성 맵을 다운샘플링하여 특성 맵의 크기를 줄이는 연산
- 학습해야 할 가중치가 없으며 연산 후에 채널 수가 변하지 않는다.

- 풀링의 종류

(1) 최대 풀링(max pooling)

- 합성곱 연산과 마찬가지로 계산
- 필터와 겹치는 영역 중 가장 큰 값을 추출하는 방식

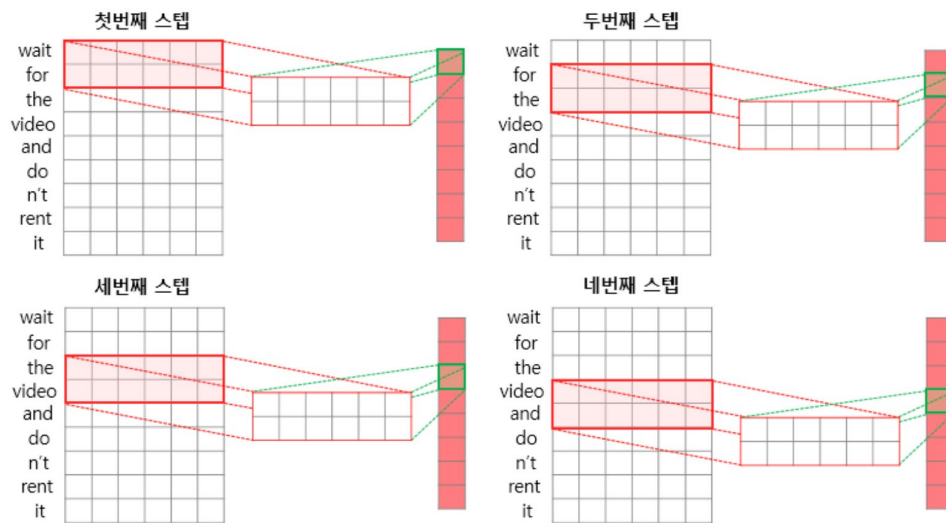
(2) 평균 풀링(average pooling)

- 필터와 겹치는 영역의 평균 값을 추출 하는 방식

6) Conv

(1) 1D

- 행렬 $k \times n$ (k : 임베딩 열벡터, n : 문장의 길이 행벡터)
- 필터(커널)의 너비는 문장 행렬에서의 임베딩 벡터의 차원과 동일하게 설정
- 커널의 크기는 가중치의 크기가 달라지므로 학습하게 되는 파라미터의 수가 달라짐



(2) 2D

- 위에 설명

7) Keras MLP 사용

(1) 초기 설정

(2) 데이터 가져오기

(3) Sequential model 구축

- `add()`
- `sequential()`

(4) 모델 알아보기

(5) 모델 컴파일하기

- `Compile()`

(6) 모델 학습하기

- `fit()`

(7) 학습 시각화 하기

(8) 테스트 데이터로 평가하기

- `Evaluate()`

(9) 예측하기

- Predict()