

딥러닝과 CNN 이해

- 다중 신경망의 근본적 문제점: 계층의 개수가 커지면 **기술기 소실** 또는 **기술기 폭증** 현상 발생
- 2000년 힌튼 교수가 심층신경망의 학습 가능성 데모
- 초기 딥러닝의 기반 모델:

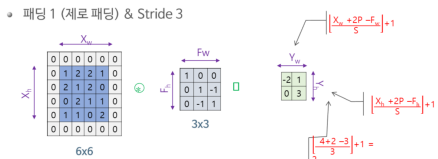
- **컨볼루션망:**
 - * 여러 노드가 가중치 공유를 통한 **계산량 축소**
 - * 위치와 위상적 특징 **추출 효율화**
 - * **서브 샘플링으로 공간 축소**
- **순환망:**
 - * **일종의 상태(memory) 유지 효과**

- 딥러닝 활성화 배경:
 - 혁신 알고리즘
 - 공개 개발환경
 - 고성능 컴퓨팅 환경(GPU, Cloud)
 - 대규모 데이터셋

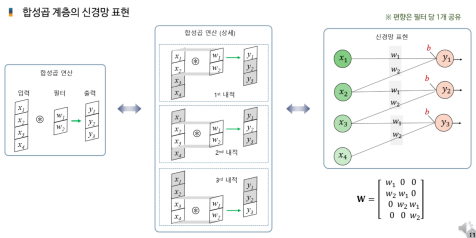
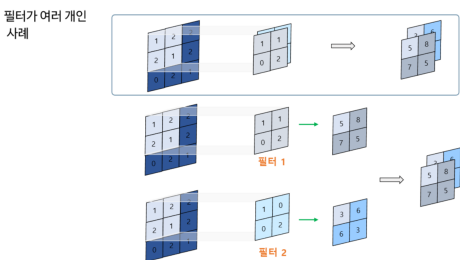
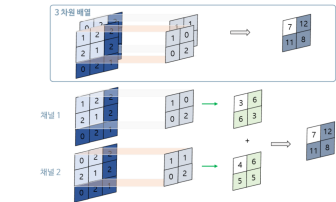
- 컴퓨터 비전:
 - 전통적인 컴퓨터 비전: **막대한 수작업** 처리 필요
 - 딥러닝: **자동 특징 추출** 가능
 - 연산 유형: cross-correlation, **convolution**

- CNN:
 - CNN은 오랜 역사를 지니고 있으며, 최근 딥러닝의 붐을 불러일으키는데 **중추 역할**
 - 컴퓨팅 필요량의 최소화를 통해 주로 **영상/비전 영역**에 큰 장점을 지님
 - **부분 연결 방식의 계층 사용, 필터를 학습 대상으로 삼음**
 - 주로 **convolution, pooling, fully connected layer**로 구성

패딩(P)은 출력을 증가, 스트라이드(S)는 출력 감소시키는 효과

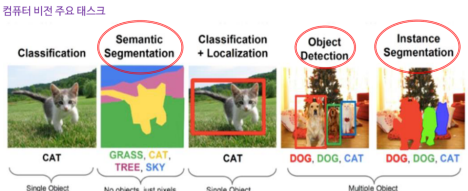
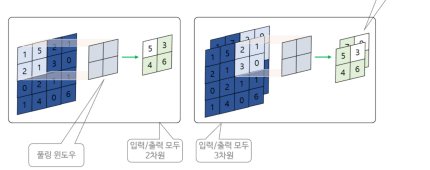


입력은 다차원 가능 (채널수 다수), 필터는 입력과 항상 동일 형상, 필터당 산출되는 출력의 채널은 항상 1개



- 결국 필터가 학습 대상이 된다.
- pooling:
 - **subsampling**이라고도 부름.
 - 일부 특징의 **이동, 잡음, 왜곡 현상에 대한 강인성을 향상**시키는 효과

- 출력 특징맵의 크기를 줄이는데 목적이 있음 (Subsampling)
- 학습 파라미터가 **없음**
- 입력 특징맵과 출력 특징맵의 차원이 항상 동일
- 일반적으로 **스트라이드(Stride)는 풀링 윈도우(풀링 윈도우)의 크기와 동일**

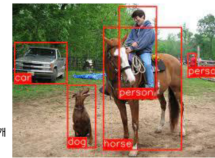


• MNIST 데이터셋

- 손글씨 숫자 이미지 데이터셋
- 28x28 픽셀, 10개의 클래스(0-9)
- 60000개의 훈련 이미지, 10000개의 테스트 이미지

■ 사물 인식(object detection)

- 객체 인식, 객체 인식이라는 이름으로도 불림
- 사진 속의 다수의 이미지 인식
- 정밀한 학습 데이터 구축이 필요
 - **바운딩 박스(bounding box)**



- 주요 시스템
 - R-CNN(2013), Fast R-CNN, Faster R-CNN, (Mask R-CNN, Mesh R-CNN)
 - ... **YOLOv5** (2020), YOLOv7(2023) 등 수십 개 모델이 등장
 - 현재도 새로운 모델의 개발은 지속적으로 인식성능 및 속도 개선 중

CVPR 2014. "Rich feature hierarchies for accurate object detection and semantic segmentation", Ross Girshick et al.

멀티 모달(multi-modal) 응용

- CNN과 RNN 등의 여러 딥러닝 모델을 포함한 기술 사용
- Image captioning, Visual question & answering 등

• 영상 주석달기(image captioning)

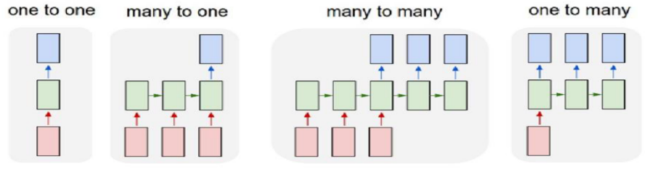
- 대표적인 학습 데이터셋: **COCO(Common objects in context), object detection, segmentation captioning dataset**



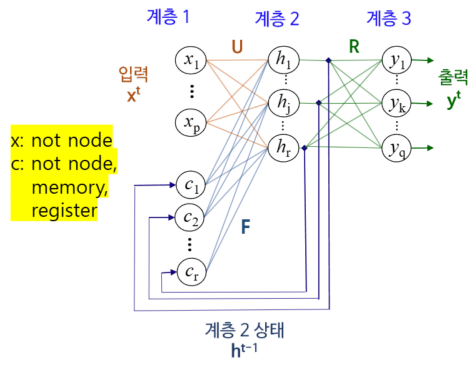
- RNN: 1 time unit delay feedback

RNN은 응용에 따라 다양한 구조를 가짐

구조 유형	입출력 개수		주요 NLP 응용	비고
	입력	출력		
일대일 one to one	1	1	언어 모델, 품사 태깅 등	각 시간계층에 1개씩의 입력과 출력
다대일 one to many	N	1	분류, 감성분석 등	N개의 입력에 1개의 출력
다대다 many to many	N	M	번역 등	Seq2seq 또는 인코더-디코더 구조
일대다 one to many	1	N	캡셔닝 등	1개의 입력에 N개의 출력



- SRN(simple recurrent network): vanilla RNN, Elman network, RNN의 대표적 모델

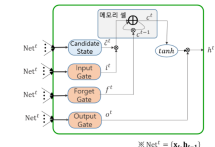


x: not node
c: not node, memory, register

LSTM 노드의 구조와 기능

- LSTM 노드는 메모리 셀과 3개의 게이트(Gate), 후보상태생성기로 구성

- **게이트**
 - 주어진 신호의 양을 조절하는 기능, 3개의 게이트 존재
 - 시그모이드 활성화함수를 사용하는 일반 뉴런
- **메모리 셀**
 - 상태(memory cell state)를 저장하는 기능
 - 게이트에 의해 그 크기가 조절됨
- **후보상태생성기**
 - 상태 생성에 사용할 후보상태를 산출 가능
- **출력**
 - 현재 상태가 tanh()를 거쳐 정규화된 이후 Output 게이트에 의해 출력 크기 가 조절

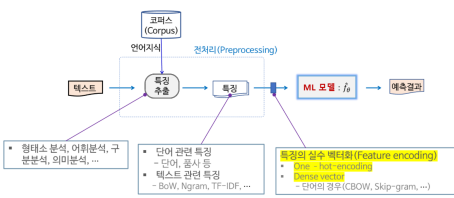


- 여러 기술기 소실 문제가 있었음.
- 자연어 처리(NLP)

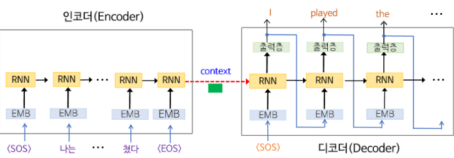
- 자연어: 인공언어와 구별하기 위해 만들어진 언어의 또다른 명칭



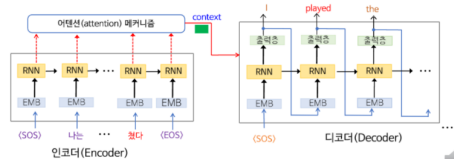
LS Linguistic Science



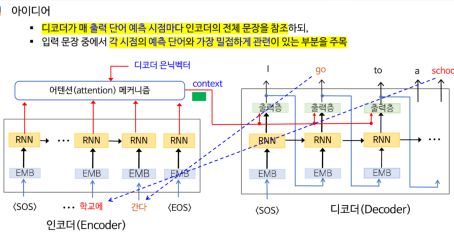
- 코퍼스: 분석에 사용하기 위해 모아놓은 텍스트 모음(코포라(corpora): 말뭉치)
 - 한국어 위키 코퍼스(kowiki), Naver 영화 리뷰 코퍼스, IMDB 영화 리뷰
 - 토큰화 코퍼스와 코드화 코퍼스가 있음
- 임베딩 방법: word2vec, GloVe, FastText 등
- SOTA: Transformer 기반
 - GPT: 트랜스포머의 디코더 사용
 - BERT: 트랜스포머의 인코더 사용
 - 트랜스포머는 Attention 기술을 근간으로 함



• 기존 seq2seq 모델 구조

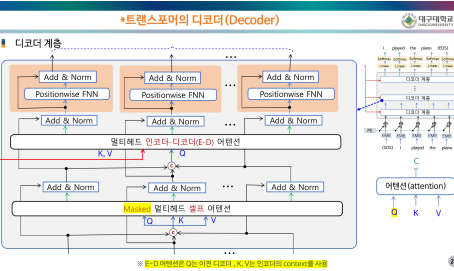


• attention 채용 seq2seq 모델 구조



- Q: 디코더의 특정시점 상태를 사용해 제작
- K,V: 인코더의 모든 은닉 상태를 사용해 제작
- 트랜스포머는 seq2seq 모델과 달리 RNN을 사용하지 않음.

- 입력 시퀀스 동시 처리, 토큰 위치 정보(positional encoding) 추가 사용



- GAN: 2-player의 경쟁(minmax 게임)에 입각한 학습
 - player1: 생성자, player2: 식별자

2-players의 훈련 방향

- 진짜 샘플과 가짜 샘플에 대한 D와 G의 훈련 방향

