

9주차 조별보고서 (Default)	
작성일: 2019년 11월 1일	작성자: 김영연
조 모임 일시: 2019년 10월 31일	모임장소: 학교 앞 카페
참석자: 김영연, 이재은, 이충현, 위성조, 최진성	조원: 김영연, 이재은, 이충현, 위성조, 최진성
구 분	내 용
학습 범위와 내용	4.1 딥러닝의 등장 4.2 깊은 다층 컨볼루션 4.3 컨볼루션 신경망 4.4 컨볼루션 신경망 사례연구
논의 내용	Q1 ResNet구조에서 네트워크 설계 시 Plain Network와 Residual Network으로 나뉘어지는데 Residual Network는 왜 잘되는지, Ensemble과 관련되어 있는지 궁금합니다.
	A1 네트워크 구조가 Ensemble인 것도 좋은 성능을 내는데 일조하지만, 네트워크가 Ensemble 구조가 되는 원인인 skip

connection이 주요한 성능 향상 기법으로 보인다.

skip connection 덕분에 gradient vanishing 문제도 최소화되어서 네트워크의 학습 속도도 더 빨라지며, 상기한 Ensemble 구조로 전체 네트워크의 성능도 더 올라가는 것으로 보인다.

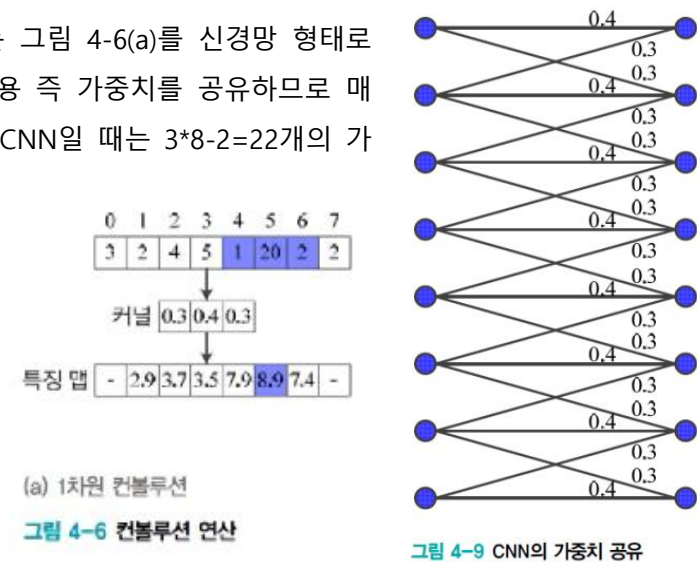
Q2

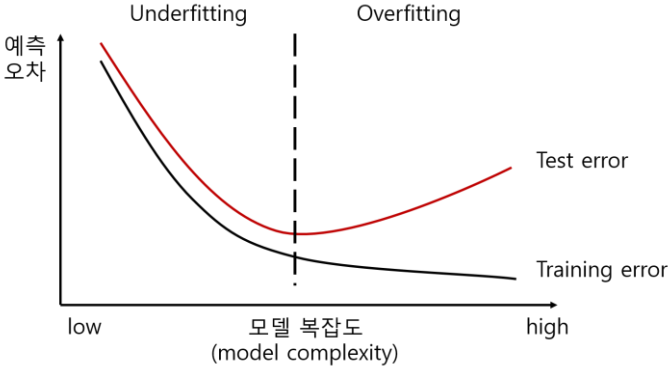
강의에서 컨볼루션 연산 시 모든 노드가 같은 가중치를 공유하므로, 모델의 복잡도가 낮아져 오버피팅이 더 쉽게 일어난다고 하셨는데, 모델의 복잡도가 낮아지면 모델의 용량이 줄어든다는 의미이고, 오버피팅보다는 언더피팅이 일어나기 쉬워지는 것이 아닌가 하는 생각이 드는데, 어째서 오버피팅이 일어날 가능성이 올라가는지 모르겠습니다.

A2

강의 4강_1 44:40 구간]을 참고하여 내용을 정리해보면 그림 4-9는 그림 4-6(a)를 신경망 형태로 바꾼 것이다. 모든 노드는 (0.3, 0.4, 0.3) 이라는 동일한 커널을 사용 즉 가중치를 공유하므로 매개변수는 3개에 불과하다. DMLP였다면 $8 \times 8 = 64$ 개의 가중치이지만, CNN일 때는 $3 \times 8 - 2 = 22$ 개의 가중치를 가지고 있으므로 모델의 복잡도가 크게 낮아진다.

이전에 배웠던 오버피팅 발생의 근본적인 원인은 학습 데이터의 부족으로, 학습데이터가 충분하지 않은 우 모델 복잡도와 오버피팅의 발생이 연결되는 것이었다. 이에 관련하여 모델 복잡도와 예측 오차 값으로 언더피팅과 오버피팅을 구분하는 그래프는 아래와 같다.



	 <p>찾아본 내용을 바탕으로는 모델 복잡도가 낮아지면 언더피팅이 발생할 확률이 높아진다는 결론을 얻었다.</p>
질문 내용	<p>강의 내용에서 컨볼루션 연산 시 모든 노드가 같은 가중치를 공유하므로, 모델의 복잡도가 낮아져 오버피팅이 더 쉽게 일어난다고 하셨고, 이에 의문점을 가져 찾아본 결과 모델 복잡도가 낮아지면 언더피팅이 발생할 확률이 높아진다고 결론을 내렸다. 이것이 맞는 답인지, 아니면 다른 이유가 존재하는지에 대해 피드백을 받고 싶습니다.</p>
기타	

첨부 개별 레포트

이름	이충현	학번	201402665
구분	내용		
학습 범위	<p>기계학습 4장</p> <p>4.1 딥러닝 기술 혁신 요소</p> <p>4.2 깊은 다층 퍼셉트론</p> <p>4.3 컨볼루션 신경망 및 풀링 연산</p> <p>4.4 컨볼루션 신경망의 발전</p>		
학습 내용	<p><4.1 딥러닝 기술 혁신 요소></p> <ul style="list-style-type: none"> ● CNN이 딥러닝의 가능성을 증폭 인터넷으로 인한 학습 데이터 늘림 더 좋은 활성화함수 오버피팅을 방지하는 효과적인 규제 기법 ● 현대 기계 학습(딥러닝) 특징 추출을 학습으로 설계 = 특징 학습 원래 패턴이 신경망의 입력 = 통째 학습 end-to-end learning 특징 학습= 앞 단계 은닉층은 저급 특징, 뒷단계 은닉층은 추상적인 형태의 고급 특징 추출. 특징 학습이 강력해짐에 따라 기존 응용에서 획기적인 성능 향상 분류나 회귀뿐만 아니라 생성 모델이나 화소 수준의 영상 분할 <p><4.2 깊은 다층 퍼셉트론></p> <ul style="list-style-type: none"> ● DMLP의 가중치 행렬 퍼셉트론(계단함수 MSE의 제곱)->다층 퍼셉트론(시그모이드, MSE의 제곱)->깊은 다층 퍼셉트론(relu 교차 엔트로피) 		

DMLP의 부진으로 인한 CNN의 부상->MNIST

< 4.3 CNN >

- DMLP vs CNN

DMLP는 완전연결 구조로 높은 복잡도

학습이 매우 느리고 과잉적합 우려

CNN은 컨볼루션 연산을 이용한 부분연결 구조로 복잡도 크게 낮춤

컨볼루션 연산은 좋은 특징 추출

영상, 음성과 같은 격자구조를 갖는 데이터에 적합

가변 크기의 입력처리 가능

- 컨볼루션의 연산은 연속함수, 이산함수의 선형연산이다.

교환법칙 성립

선형시불변시스템 = 입력 $x(t)$ 와 시스템의 임펄스 응답 $h(t)$ 와 출력 $y(t)$ 의 관계

$$Y(t) = X(t) * h(t)$$

시간에 따라 불변, 미래를 예측 가능

- 컨볼루션층에서 zero padding = 가장자리에서 입력이 줄어드는 효과 방지

가중치 공유 = 모든 노드가 동일한 커널을 사용, 가중치를 공유하므로 모델의 복잡도가 크게 낮아짐.

다중 특징 맵 추출

수직방향, 수평방향 에지 추출.

특징학습 = 커널을 사람이 설계하지 않고, 학습으로 알아냄.

ex) 2차원 영상이 7*7 커널을 64개 사용한다면, 학습은 $(7*7+1)*64 = 3200$ 개의 매개변수를 찾아내야 한다.

오류역전파로 학습한다.

- 컨볼루션 연산에 따른 CNN의 특성

	<p>1. 이동에 동변(신호가 이동하면 이동 정보가 그대로 특징 맵에 반영) -> 영상 인식에서 물체 이동이나 음성 인식에서 발음 지연에 효과적으로 대처가능하다.</p> <p>2. 병렬분산 구조 각 노드는 독립적으로 계산 가능하므로 병렬구조 노드는 깊은 층을 거치면서 전체에 영향을 미치므로 분산구조</p> <p>3. 큰 보폭에 의한 다운샘플링 일반적으로 보폭이 k이면 k개마다 하나씩 샘플링하여 커널 적용 -> 2차원 영상의 경우 특징 맵이 $1/k^2$으로 작아짐</p> <p>4. 텐서의 적용 3차원 이상의 구조에도 적용 가능. 3-d convolution</p> <ul style="list-style-type: none"> ● 풀링층 최대 풀링, 평균 풀링, 가중치 평균 풀링 등 존재. 보폭을 크게 하면 다운샘플링 효과 발생 특성 = 매개변수가 없음, 특징 맵의 수를 그대로 유지 작은 이동에 둔감 = 물체 인식 영상 검색 등에 효과적이다. 전체 구조 = 컨볼루션층->활성 함수(relu)->풀링층 다중 커널을 사용하여 다중 특징 맵을 추출한다. ● 전체구조 초창기에는 특징추출 = c-p-c-p-c의 다섯 층, 28*28 맵을 120차원의 특징 벡터로 변환 분류 = 은닉층이 하나인 MLP 가변크기의 데이터 다루기 = CNN은 가능, 보폭 조정하여 특징 맵 크기 조절 <p><4.4 절 컨볼루션 신경망 사례연구> ImageNet 데이터베이스로 자연영상 분류라는 도전적 문제를 내세웠다.</p>
--	---

	<ul style="list-style-type: none"> ● AlexNet 컨볼루션층 5 개(200 만개의 매개변수)와 완전연결층 3 개(6500 만개의 매개변수), 총 8 개 층 성공요인 = imagenet, gpu 사용한 병렬처리, relu, 정규화 기법(normalization), 오버피팅 방지하는 여러 규제기법(regularization)적용 ● VGGNet 커널을 3*3 쓰는 대신 신경망을 더욱 깊게 만듦, 컨볼루션층 8-16 개를 두어 AlexNet 에 비해 2-3 배 깊어짐 1*1 커널을 처음 도입 = 차원 축소 효과, relu 같은 비선형 활성화함수를 적용하면 특징 맵의 분별력이 증가한다. 그러나 최종선택하지는 않음 ● GoogleNet 인셉션 모듈 = NIN 의 구조를 수정한 것. 1*1 컨볼루션을 사용하여 차원 축소. MLPconv 대신 4 종류의 컨볼루션 연산 사용. NIN 의 구조 = MLPconv 층이 컨볼루션 연산을 대신한다. 커널을 옮겨가면서 MLP 의 전방 계산을 수행한다. 전역 평균 풀링사용 = MLPconv 가 부류 수만큼 특징 맵을 생성하면, 특징 맵 각각을 평균하여 출력 노드에 입력하므로서 매개변수들을 없앤다(과잉 적합 문제도 해결됨) ● ResNet 잔류학습이라는 아이디어를 이용하여 성능 저하를 피하면서 층 수를 대폭 늘림. 지름길 연결을 하는 이유는 그래디언트 소멸 문제를 해결하고자 하기 위함이다. 3*3 커널 사용, 잔류학습 사용, 전역 평균 풀링 사용(FC 층 제거), 배치 정규화 적용(드롭아웃 적용 불필요) 		
질문 내용	ResNet 구조에서 네트워크 설계시 Plain Network 와 Residual Network 으로 나뉘어지는데 Residual Network 는 왜 잘될까? Ensemble 과 관련되어 있을까?		
이름	김영연	학번	201500629

구분	내용
학습 범위	<p>4.1 딥러닝의 등장</p> <p>4.2 깊은 다층 퍼셉트론</p> <p>4.3 컨볼루션 신경망</p> <p>4.4 컨볼루션 신경망 사례연구</p>
학습 내용	<ul style="list-style-type: none"> ● 딥러닝의 기술 혁신 요인 <ol style="list-style-type: none"> 1. CNN <ul style="list-style-type: none"> -작은 크기의 컨볼루션 마스크 사용 -> 매개변수가 훨씬 적다. - 가중치 공유 기법 사용 ->모든 노드가 같은 마스크 공유 2. 값싼 GPU <ul style="list-style-type: none"> -손쉽게 병렬처리 가능, 학습시간 단축 3. 학습데이터의 증가 4. 계산은 간단하고 성능은 좋은 활성화함수 개발 <ul style="list-style-type: none"> - ReLU 함수는 $\text{ReLU}(a)=\max(0,a)$이므로 비교연산 한 번으로 계산할 수 있고, 미분값은 0 또는 1이므로 역시 비교 한 번으로 계산할 수 있다. 5. 다양한 규제기법 등장 <ul style="list-style-type: none"> -가중치 축소기법 -> 가중치를 작은 값으로 유지하는 기법 -드롭아웃 기법 -> 임의로 일정 비율의 노드를 선택하여 불능으로 놓고 학습하는 기법 6. 층별 예비학습 기법

- 통째학습(end-to-end learning)

-깊은 신경망에서는 특징추출이라는 작업 자체도 기계학습으로 설계한다. 여러 단계의 은닉층은 자동으로 계층적인 특징을 추출하는데, 앞 단계의 은닉층은 주로 에지나 코너 같은 저급 특징(low-level feature)을 추출하고 뒤로 갈수록 더 추상적인 형태의 고급 특징을 추출하는 것.

이와 같이 전체 과정을 학습하는 방식을 통째학습이라고 한다.

- DMLP와 CNN의 비교

- DMLP는 완전연결 구조여서 가중치가 높다: 왼쪽 층의 모든 노드와 오른쪽 층의 모든 노드가 연결된 형태

두 층이 각각 a개와 b개의 노드를 가진다면 총 가중치는 a*b

CNN은 부분연결 구조: 오른쪽 j번째 노드는 왼쪽 층의 i-1, i, i+1위치에 있는 노드 3개와 연결

오른쪽 층이 a개의 노드를 가진다면 왼쪽 층의 노드 개수와 상관없이 가중치는 3*a

- DMLP에 입력되는 데이터는 일렬로 늘어선 벡터 구조이고 입력 샘플이 항상 같은 크기여야 한다.

CNN은 영상과 같은 행렬 구조 또는 3차원 이상의 텐서 구조까지 처리할 수 있고 샘플의 가변 크기 입력까지 처리할 수 있다.

- CNN은 격자구조를 가진 데이터에 적합하고, 부분연결을 구현하기 위해 컨볼루션 연산을 적용한다. 또한, 컨볼루션 연산을 적용하여 얻은 특징 맵을 다운 샘플링하기 위해 풀링 연산을 이어 수행한다.

- 컨볼루션층

1. 컨볼루션 연산

- 컨볼루션은 해당하는 요소끼리 곱하고 결과를 모두 더하는 단순한 선형연산이다.

$s(i) = \sum_{x=-(h-1)/2}^{(h-1)/2} z(i+x)u(x)$ h는 커널의 크기이고, 출력 s를 특징 맵이라고 한다.

-2차원 컨볼루션

$s(i,j) = \sum_{y=-(h-1)/2}^{(h-1)/2} \sum_{x=-(h-1)/2}^{(h-1)/2} z(i+y, i+x)u(y,x)$ 로 나타낼 수 있다. 이때 문제점은 데이터의 가장자리에서는 커널이 데

이더 영역의 바깥쪽으로 나가게 되므로 연산이 불가능하고 이로 인해 줄어드는 양이 많아지는 것이다. 이를 해결하기 위하여 덧대기를 수행한다.

-바이어스: CNN도 바이어스가 존재한다. 바이어스는 항상 1을 입력 받으므로 컨볼루션 결과에 바이어스 값만큼 더하는 효과가 있다.

2. 가중치와 공유와 다중 특징 맵 추출

-가중치 공유: 같은 커널을 사용하므로 가중치를 공유

-서로 다른 커널은 다른 형태의 특징을 추출하고 실제로는 수십~수백개의 커널을 사용하여 다양한 특징 맵을 추출한다.

- 다중 특징 맵의 주목할 점

1) 커널의 크기가 3이어도 커널의 실제 크기는 1*4라는 점-> 바이어스 노드를 맨 왼쪽에 추가한 것

2) 커널의 요소를 u_i^k 로 표시-> k번째 커널의 i번째 요소

3. 컨볼루션 연산에 따른 CNN의 특성

1) 컨볼루션은 이동에 동변이다. -> 신호가 이동하면 이동 정보가 특징 맵에 그대로 반영된다.

2) 병렬 분산 구조 -> CNN은 깊은 구조를 가지므로 영향력 범위가 층을 거듭할수록 점점 커져 뒤로 가면 맵 전체에 영향을 준다.

● 풀링층의 특성

1. 학습할 매개변수가 없다는 점 -> 단지 최대 또는 평균을 구하는 연산을 하기 때문에 학습이 알아내야 할 매개변수가 없다.

2. 특징 맵의 수가 그대로 유지된다는 점 -> 특징 맵마다 독립적으로 풀링 연산을 적용하므로 특징 맵의 개수가 유지

3. 작은 이동에 둔감해지게 하는 특성

질문 내용	
-------	--

이름	위성조	학번	201402033
구분	내용		
학습 범위	4.1 딥러닝의 등장 4.2 깊은 다층 퍼셉트론 4.3 컨볼루션 신경망 4.4 컨볼루션 신경망 사례연구		
학습 내용	1980년대에 이미 깊은 신경망 아이디어가 있었으나, 그레이디언트 소멸 문제, 훈련 데이터 셋이 많지 않음, 컴퓨팅 능력 부족으로 과다한 계산 시간 등의 이유로 실현되지 않았다. 학습률에 따른 성능 변화, 모멘텀의 영향, 은닉 노드 수에 따른 성능 변화, 데이터 전처리, 활성화함수, 규제 기법등의 변화로 성능 이 점점 개선되었다. 기계 학습의 패러다임 변화 -고전적인 다층 퍼셉트론		

은닉층은 특징 추출기/ 사람이 직접 특징을 추출하여 신경망에 입력함

-현대 기계 학습(딥 러닝)

특징 추출을 학습으로 설계/원래 패턴이 신경망의 입력(end-to-end learning)



특징 학습(feature learning 또는 표현 학습(representation learning))

앞 단계 은닉층은 에지나 코너와 같은 저급 특징 추출

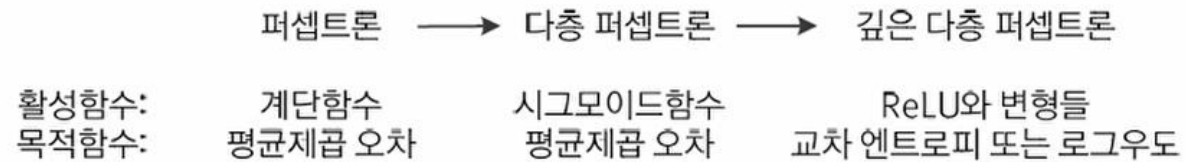
뒷 단계 은닉층은 추상적인 형태의 고급 특징을 추출

특징 추출 후 분류 단계에서는 주로 한 층, 혹은 두 층 정도의 fully-connected layer를 사용

DMLP(Deep Multi-Layer Perceptron) 학습은 MLP 학습과 비슷, 단지 더 많은 단계에 걸쳐 수행

$l + 1$ 번째 층의 정보를 이용하여 l 번째 층의 그레이디언트 계산

학습 알고리즘의 주요 개선



DMLP

완전연결 구조로 높은 복잡도

학습이 매우 느리고 과잉적합 우려가 있음

CNN

컨볼루션 연산을 이용한 부분연결(희소연결) 구조로 복잡도 크게 낮춤

컨볼루션 연산은 좋은 특징 추출

격자 구조(영상, 음성 등)를 갖는 데이터에 적합

수용장(receptive field)은 인간 시각과 유사

가변 크기의 입력 처리 가능

CNN이 고성능을 내기 위해서는 receptive field 간에 correlation이 높아야 한다.

컨볼루션 연산

$$(f * g)(t) \triangleq \int_{-\infty}^{\infty} f(\tau)g(t - \tau) d\tau. \quad (f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]g[n - m]$$

컨볼루션 연산은 교환법칙이 성립하여

$$\int_{-\infty}^{\infty} f(\tau)g(t - \tau) d\tau = \int_{-\infty}^{\infty} g(\tau)f(t - \tau) d\tau \text{ 가 성립한다.}$$

선형시불변시스템(LTI, Linear Time Invariant system) – 시간에 의해 결과값이 변하지 않는 시스템으로서, 입력 $x(t)$ 를 알면 미래 결과를 예측할 수 있다.

컨볼루션은 해당하는 요소끼리 곱하고 결과를 모두 더하는 선형 연산

CNN에서 학습시켜야 하는 곳은 커널.

예전에는 사람이 직접 커널의 값을 결정했었음.

가장자리에서 입력이 줄어드는 효과를 방지하기 위해 0 덧대기(zero padding)와 복사 덧대기를 사용하기도 한다.

바이어스항 추가 (실제로는 영상을 노멀라이제이션 하는 등의 이유로 잘 쓰이지 않으나, 바이어스 항을 추가하기도 한다.

모든 노드가 동일한 커널을 사용(가중치를 공유)하므로 모델의 복잡도가 크게 낮아진다.

커널의 값에 따라 커널이 추출하는 특징이 달라진다. 따라서 하나의 커널만 사용하면 너무 빈약한 특징이 추출되어 여러 개(수십 ~수백 개)의 커널을 사용하게 된다.

$$\begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \text{ 수직방향, } \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} \text{ 수평방향 에지 추출}$$

특징 학습 - 커널을 사람이 설계하지 않고, 학습으로 알아냄

CNN의 특성

신호가 이동하는 정보가 그대로 특징 맵에 반영됨 -> 영상 인식에서 물체 이동이나 음성인식에서 발음 지연에 효과적으로 대처 가능

병렬분산 구조 -> 각 노드는 독립적으로 계산 가능하므로 병렬 구조

노드는 깊은 층을 거치면서 전체에 영향을 미치므로 분산 구조

큰 보폭에 의한 다운샘플링

모든 화소에 커널 적용 -> 보폭을 1로 설정한 것과 같음.

일반적으로 보폭이 k 이면 k 개 마다 하나씩 샘플링하여 커널 적용 -> 2차원 영상의 경우 특징 맵이 $1/K^2$ 로 작아짐

3차원 이상의 구조(텐서)에도 적용 가능

풀링 연산

최대 풀링, 평균 풀링, 가중치 평균 풀링 등 / 보폭을 크게 하면 다운샘플링 효과

풀링 연산의 특성

풀링은 상세 내용에서 요약통계를 추출함

매개변수가 없음 = 학습시킬 가중치가 없음.

특징 맵의 수를 그대로 유지함

작은 이동에 둔감 -> 물체 인식이나 영상 검색 등에 효과적임

CNN은 빌딩 블록을 이어 붙여 깊은 구조를 만든다.

활성함수로 주로 ReLU를 사용한다.

LeNet-5

C-P-C-P-C의 다섯층을 이용./ 은닉층이 하나인 MLP / CNN의 첫 번째 성공사례 : 필기 숫자 인식기로 수표인식 시스템 구성.

DMLP와는 다르게 CNN은 가변 크기 특징 벡터를 다룰 수 있다

- 컨볼루션층에서 보폭 조정, 혹은 풀링층에서 커널이나 보폭을 조정하여 특징 맵 크기를 조절할 수 있다.

ImageNet 데이터베이스 : 2만 부류에 대해 부류별로 500~1000장의 영상을 인터넷에서 수집하여 구축하고 공개

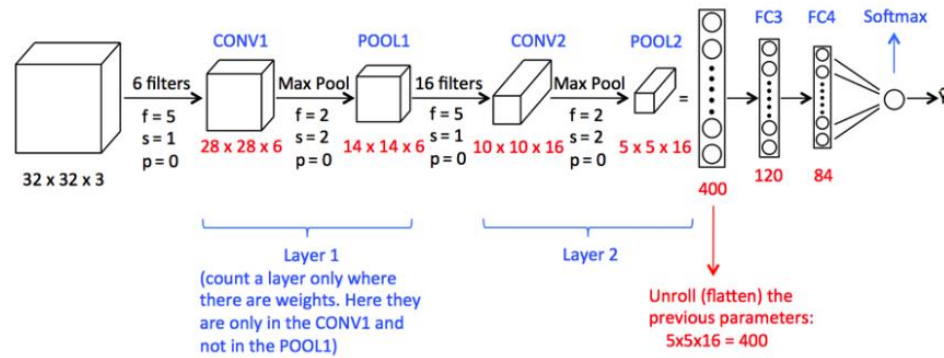
ILSVRC 대회 : 1000부류에 대해 분류, 검출, 위치 지정 문제 //120만장의 훈련집합, 5만장의 검증집합, 15만장의 테스트집합

이름	AlexNet	VGGNet	GoogLeNet	ResNet(2014년 버전)
특징	컨볼루션층 5개, 완전연결(FC)층 3개	컨볼루션층 13개, 완전연결층 3개 3*3의 작은 커널을 사용하여 신경망을 더욱 깊게 만들	NIN의 구조를 수정 매개변수가 있는 층 22개, 없는 층 5개로 총 27개 층. 완전연결층은 1개에 불과	잔류 학습 이용 잔류 학습은 지름길 연결된 x 를 더한 $F(x)+x$ 에 τ 를 적용 -지름길 연결을 두는 이유 그레이디언트 소멸 문제 해결
특이사항	ImageNet이라는 대용량 데이터베이스 GPU를 사용한 병렬처리 활성함수로 ReLU 사용 지역 반응 정규화 기법 적용 여러 규제 기법 적용(데이터 확대, 드롭아웃 등) 테스트 단계에서 앙상블 적용	1*1 커널 - 차원 축소 효과 Ex) 8*m*n 텐서에 8*1*1 커널을 4개 적용하여 4*m*n 텐서 출력 VGGNet은 1*1커널을 실험하였으나, 최종 적용하지는 않았음	전역 평균 풀링 MLPconv가 부류 수 만큼 특징 맵을 생성하면, 특징 맵 각각을 평균하여 출력 노드에 입력 -> 매개 변수가 사라짐. 인셉션 모듈 - 마이크로 네트워크로 MLPconv 대신 네 종류의 컨볼루션 연산 사용 1*1 컨볼루션 사용	3*3 커널 사용 잔류 학습 사용 전역 평균 풀링 사용(FC층 제거) 배치 정규화 적용 2015년 개선판에서는 $F(x)+x$ 에 ReLU 적용

	<p>NIN(Network In Network)</p> <p>MLPconv층이 컨볼루션 연산을 대신함.</p> <p>MLPconv는 커널을 옮겨가면서 MLP의 전방 계산을 수행</p>
질문 내용	<p>강의에서 컨볼루션 연산 시 모든 노드가 같은 가중치를 공유하므로, 모델의 복잡도가 낮아져 오버피팅이 더 쉽게 일어난다고 하셨는데, 모델의 복잡도가 낮아지면 모델의 용량이 줄어든다는 의미고, 오버피팅보다는 언더피팅이 일어나기 쉬워지는 것이 아닌가 하는 생각이 드는데, 어째서 오버피팅이 일어날 가능성이 올라가는지 모르겠습니다.</p>

이름	이재은	학번	201502469
구분	내용		
학습 범위	<p>4.1절: 딥러닝을 성공으로 이끈 기술 혁신 요소</p> <p>4.2절: 다층 퍼셉트론의 구조와 학습 알고리즘</p> <p>4.3절: 컨볼루션과 풀링 연산</p> <p>4.4절: 컨볼루션 신경망의 발전</p> <p>4.5절: GAN</p> <p>4.6절: 딥러닝이 강력한 이유</p>		

<p>학습 내용</p>	<ul style="list-style-type: none"> - 딥러닝의 특징: 특징 추출을 학습으로 설계하는 특징학습, 원래 패턴이 신경망의 입력이 되는 통째학습 - 특징? 컴퓨터가 사진을 보고 물체를 인식하기 위해서는 사진에서 물체를 판단하는데 필요한 특징들을 검출하여야 한다. 이러한 특징을 사람이 직접 설계한 뒤, 각각의 물체의 사진에서 나타나는 특징들의 차이점을 가지고 학습하여 사진을 이해하고 처리한다. - CNN의 특징학습: 커널을 사람이 설계하지 않고 학습으로 알아낸다. - 오류 역전파는 역방향으로 오차를 전파시키면서 각층의 가중치를 업데이트 하고, 최적의 학습 결과를 찾아가는 방법이다. 단계는 1. 그레디언트 계산- 2. 가중치 갱신 순이다. - 활성화함수: 뉴럴 네트워크의 개별 뉴런에 들어오는 입력신호의 총합을 출력 신호로 변환하는 함수 - DMLP: 완전연결 구조로 높은 복잡도, 학습이 매우 느리고 과잉적합 우려 CNN: 컨볼루션 연산을 이용한 부분연결(희소 연결) 구조로 복잡도 크게 낮춤 - 선형 시불변 시스템 (LTI, Linear Time Invariant System): 선형성(중첩의 원리) 및 시불변성(Time-invariant) 특성을 둘다 갖는 시스템 - 컨볼루션 연산의 성질은 다양한 곳에 많이 사용되니 잘 알아두자. - CNN 사례 두 단의 convolution-pooling 과정과 두 단의 fully connected layer, output layer 로 구성. CONV 층은 ReLU 를 포함한 것이다.
--------------	---



f: filter size, s: stride, p: padding (상하좌우에 0 을 채워 넣는 것. p=0 이면 padding 을 하지 않았다는 의미임)

Softmax: sigmoid 구조에 정규화과정을 추가하여 출력단의 합이 1 이 되도록 만든 계층. 출력단으로 주로 사용함

- 덧대기(Zero Padding): 컨볼루션 연산 결과 크기가 줄어드는 것을 방지하기 위해 가장자리에 0 을 집어 넣는 것
- 풀링(pooling): 이미지 특징을 반영하며, 차원을 축소
- 풀링(pooling)의 종류: 1. Global average pooling: window size 나 stride 를 지정해 주지 않고 feature map 상의 노드 값들의 평균을 뽑아내서 CNN 의 차원을 줄인다. 2. Max pooling layer: window size 와 stride 를 지정해 준 후 window 상에서 포함하고 있는 픽셀들 중 최대값을 뽑아낸다.
- 드롭아웃: 인공 신경망의 뉴런을 확률적으로 사용하지 않음으로써 과적합을 방지하는 기법이다.

질문 내용

이름	최진성	학번	201403474
구분	내용		
학습 범위	기계학습 4장 딥 러닝 4.1 딥 러닝의 등장 4.2 깊은 다층 퍼셉트론 4.3 컨볼루션 신경망 및 풀링 연산 4.4 컨볼루션 신경망 사례 연구		
학습 내용	기계학습 4장 딥 러닝 4.1 딥 러닝의 등장 이미 깊은 신경망 아이디어가 있었으나 당시 기술력으로는 실현이 불가능 <ul style="list-style-type: none"> ● Gradient 소멸 ➔ 새로운 규제 비법 및 활성화함수, 예비학습 기법 ● 훈련집합의 작은 크기 ➔ 인터넷 등장(학습 데이터 증가) ● 과도한 계산 시간(컴퓨터의 성능 부족) ➔ 값싼 GPU가 등장 CNN이 등장함으로써 딥 러닝의 가능성을 열었음 특징 학습 은닉층은 '특징'을 추출하는 특징 추출기의 역할을 한다. 구조가 알기 때문에 센서로 획득한 패턴을 그대로 입력하면 성능이 낮음 -> 사람이 수작업으로 특징 구상 및 구현 기존의 학습은 원래 패턴 전체를 신경망에 입력하는 End-to-End Learning(전체 학습)		

현재는 특징 추출만을 학습으로 설계하는 특징 학습이 대두됨.

*기존에는 수동으로 설계하여 특징을 추출, 현재는 학습집합 내에서 특징을 찾아서 학습.

은닉층을 나누어서 층 별로 추출하는 특징을 분리

앞 단계 은닉층 : Edge, Corner같은 저급 특징을 추출

뒷 단계 은닉층 : 위의 형태를 혼합하거나 추상적인 형태의 고급 특징을 추출

특징 학습의 발달로 영상, 음성 등에서 뛰어난 성능을 보이게 되었으며 언어 번역 쪽에도 큰 영향을 끼치게 됨.

또한 새로운 모델을 생성하거나 화소 수준의 영상을 분할하는 등으로 발전함.

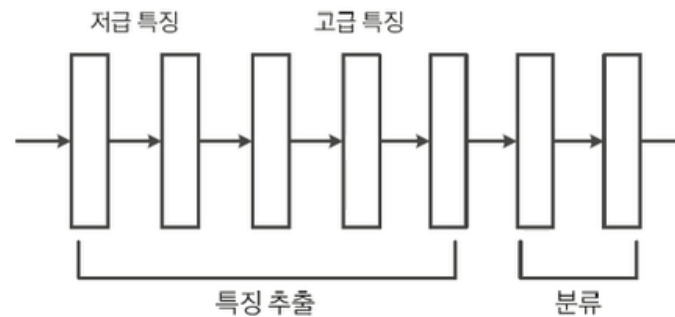


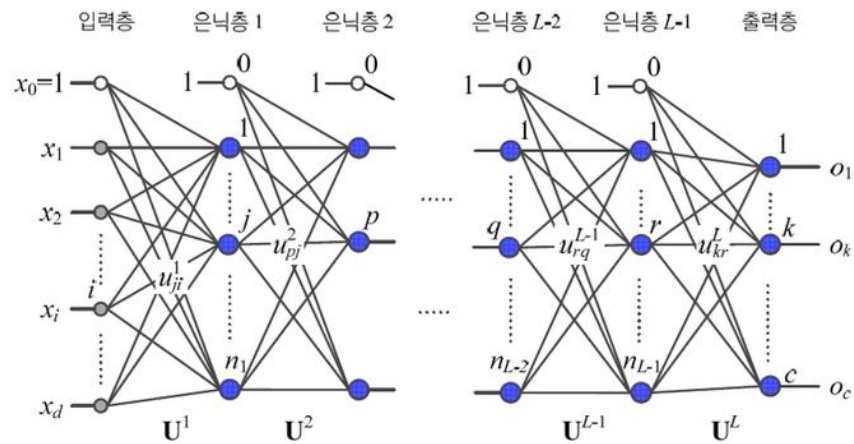
그림 4-2 깊은 신경망의 처리 절차

DMLP와 마찬가지로 오류 역전파로 커널을 학습한다.

4.2 깊은 다층 퍼셉트론(DMLP)

전체적으로 MLP와 비슷한 느낌을 가진다.

구조는 $d+1$ 개의 입력층, c 개의 출력층, $L-1$ 개의 은닉층으로 이루어져 있다.

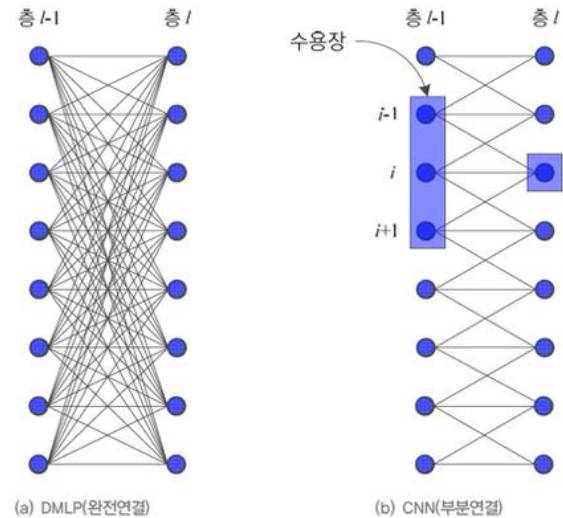


$$\mathbf{o} = \mathbf{f}(\mathbf{x}) = \mathbf{f}_L \left(\cdots \mathbf{f}_2 \left(\mathbf{f}_1(\mathbf{x}) \right) \right)$$

퍼셉트론의 역사

초기 퍼셉트론 등장(계단함수, 평균제곱 오차) -> 다층 퍼셉트론 고안(Sigmoid[Back-Propagation 고려], 평균제곱 오차)
-> 깊은 다층 퍼셉트론(ReLU 및 변형들, 교차 엔트로피 혹은 로그 우도)

4.3 CNN



DMLP는 완전 연결 구조이기 때문에 높은 복잡도를 가지며 학습이 매우 느리고 과잉적합이 빈번하다.

또한, 특징 벡터의 크기에 변동이 있을 경우 연산이 불가능하다.

CNN은 부분연결(희소 연결)을 이용해 복잡도를 크게 낮추었고 특징 추출면에서 DMLP보다 더 뛰어나다.

보통 CNN은 격자 구조의 데이터를 가진 매체에 적합하다(ex : 영상, 음성)

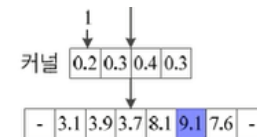
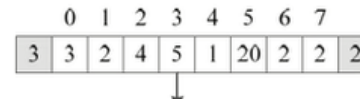
특징 벡터의 크기에 변동이 있더라도 Convolution층이나 Pooling 층에서 보폭을 조정하여 특징 맵 크기를 조절할 수 있다.

Convolution - 연속 함수간의 operation, 이산 함수간의 연산을 의미한다.

어떤 시스템이 시간에 따라서 변하지 않고 선형적인 특징을 가질 경우 미래를 예측하기 쉬운 특성을 지님.

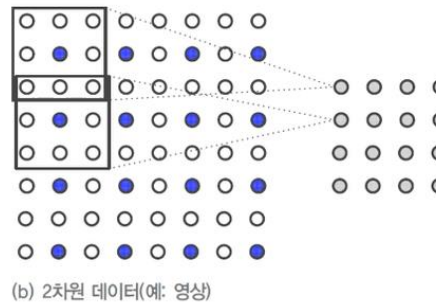
즉, 해당하는 요소끼리 곱하고 결과를 모두 더하는 선형 연산.

각 차원 별 Convolution 연산



*다중 특징 맵 추출 : 커널의 값을 바꾸는 것으로 다른 특징을 추출할 수 있음. -> 적은 자원으로 많은 특징을 추출할 수 있음.

➔ 영상, 혹은 음성 인식에서 효과적으로 학습을 대처할 수 있게 해줌.



텐서 - 3차원 형태로 쌓여 모이는 것. CNN의 경우엔 3차원 구조의 데이터 형식인 텐서에도 적용이 가능하다.

풀링

Maximum, Average, Weight Pooling

값을 비교하여 다음 층에 값을 전달.(maximum : 보폭 1개의 거리를 가진 값들 중 가장 큰 값을 다음 층에 전송)

즉, 상세 내용에서 요약통계를 추출하기 때문에 매개 변수가 없음.

특징 맵의 수를 그대로 유지하나, 작은 이동에 대해서 둔감함

빌딩 블록

CNN의 구조로, Convolution층 - 활성화함수(ReLU)층, 풀링층으로 이루어짐

다중 커널을 이용하여 다중 특징 맵을 추출함.

*초기 CNN 사례 : LeNet-5

4.4 컨볼루션 신경망 사례 연구

*자연 영상 분류

AlexNet – 이미지 특징 분류에서 획기적인 성능을 보임

활성함수로 ReLU를 사용, 지역 반응의 Regularization 기법 및 과잉적합 방지를 위한 Normalization 도입
GPU를 이용하여 병렬처리화, 앙상블 적용

VGGNet – 커널을 3*3의 작은 커널만을 이용하여 더욱 깊게 만듦.

차원을 축소하는 효과를 이용하여 ReLU와 같은 비선형 활성화함수를 통해 특징 맵의 분별력을 증가시킴

GoogLeNet – NIN 구조(MLPconv층이 Convolution 연산을 대신하는 것) 확장

* NIN 구조

- MLPconv는 커널을 옮겨다니면서 MLP의 전방 계산을 수행한다.

- Global Average Pooling 사용(FC층 제거)

-> 특징 맵 각각을 평균하여 출력 노드에 입력하여 매개변수를 줄임 -> 과잉적합 방지

GoogLeNet은 MLPconv 대신 네 종류의 Convolution 연산을 사용함 + 1*1 Convolution을 이용하여 차원 축소
완전연결층을 하나만 운용함. VGGNet에 비하면 1%에 불과한 매개변수를 가짐.

ResNet – 정보가 줄어드는 것을 막기 위해 별도로 입력을 전달하는 방식을 도입(지름길 연결)

→ Gradient 소멸 문제를 해결함

잔류 학습 사용 및 Global Average Pooling 사용,(FC층 제거)

배치 정규화 사용 및 3*3 커널 사용

경이로울 정도로 낮은 오류율 달성.

*ILSVRC 대회

	<p>분류 문제 자체는 성능 포화 수준 단계까지 올라갔으므로 사람 수준의 성능에 필적한 문제를 제시한다.</p>
질문 내용	