딥러닝 소개 및 주요 이슈

최희열·민윤홍 (삼성전자 종합기술원)

목차

1. 서 론
2. 배 경
3. 학습 원리
4. 주요 알고리즘
5. 응 용
6. 결 론

1. 서 론

인간 지능을 모사하려는 인공지능 기술은 점진적으로 발전해 왔다. 이런 인공지능 기술은 1997년 딥블루(Deep Blue) 가 체스 세계 챔피언 G. Kasparov 를 이기면서 많은 대중들의 관심을 끌었다. 이런 딥블루가 많은 대중성 측면에서는 성공했지만, 딥블루의 지능은 디지털화된 체스 고수들의 지식과 엄청나게 많은 경우의 수를 계산하는 컴퓨팅 파워에 의존하기 때문에 기술적 측면에서는 높은 평가를 받기 어려웠다. 그에 비해, IBM의 왓슨(Watson)이나 Apple의 시리(Siri), Google 나우(Now) 등으로 대표되는 최근의 인공지능 기술은 해당 분야의 전문 지식에 의존하는 대신 빅데이타에 기반하여 지식을 자동으로 축적한다는 면에서 인간 수준의 인공지능을 향하여 진일보 했다고 할 수 있다.

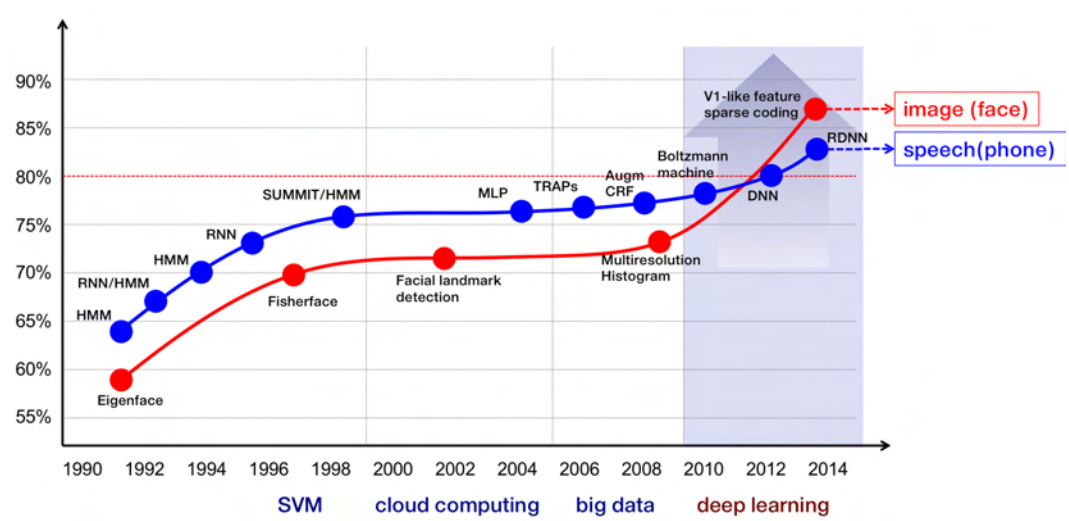
인공지능을 이해하는 여러가지 관점과 접근이 있겠지만, 최근 패턴인식 성능의 비약적인 향상을 주도해온 데이타 기반 인공지능을 살펴보자. (그림 1)에서 볼 수 있는 것처럼, 1990년대에 작은 데이타에 기반한 패턴인식은 이론적인 발전에도 불구하고, 그 성능은 인간 지능의 성능에는 크게 미치지 못했다. 하지만, 2000년대 들어서면서부터 시작된 데이타의 대용량화와 이를 처리할 수 있는 클라우드 기반의 컴퓨팅 파워의 폭발적 증가는 기존 패턴인식 분야의 알고리즘들의 한계를 극복하는 새로운 패러다임의 패턴인식을 기대하게 만들었다[2,10]. 딥러닝(deep learning)은 이런 배경해서 최근의 음성인식과 영상인식을 비롯한 다양한 패턴인식 분야의 성능향상을 이끄는 중요한 인공지능 기술이다.

딥러닝의 개념은 사실 1980년대에, 혹은 이미 그 이전부터 제안되고 논의되어 왔었다. 하지만, 2006년 사이언스에 발표된 Hinton 교수의 논문 이후에서야 많은 사람들이 딥러닝을 체계적으로 연구하기 시작했다[1,28]. 뿐만 아니라, 딥러닝에 기반한 여러 패턴인식 서비스들이 글로벌 IT 업체들에 의해 제공되면서, 딥러닝은 단순히 관련 연구자들 뿐만 아니라, 대중매체에서도 인공지능을 향한 커다란 발전으로 인식하고 관련 기사들을 쏟아내고 있다 [3-5]. 특별히 한동안 뜨거운 감자였지만, 분석하고 이해할 방법이 없었던 빅데이타에 대해 딥러닝이 분석 도구로써 작용할 수 있을 것으로 기대되면서 더욱 주목받고 있다.

기본적으로 딥러닝은 기존 신경망(neural networks) 에 계층수를 증가시킨 심층신경망(deep neural networks) 혹은 심층망(deep networks) 을 효과적으로 학습하여 패턴인식이나 추론에 활용하는 것을 말한다. 이런 심층망의 장점은 기존의 신경망에 비해 더 많은 중간 계층을 사용함으로써 데이터에 대한 표현 능력을 크게 증가시킬 수 있다는 것이다. 이러한 심층망의 아이디어는 이전부터 존재했지만, 2006년 Hinton 교수의 논문 이전에는 심층망을 위한 효과적인 학습방법의 부재로 크게 주목받지 못했다. Hinton 교수는 2006년 논문에서 사전 학습(pre-training)이라는 개념을 제안함으로써 심층망의 학습 가능성을 보여주었고, 그 이후 여러가지 다양한 학습 방법들이 제안되어 사용되고 있다[6].

신경망에서 중간계층을 여러겹 쌓는 것은 단순해 보이지만, 이러한 양적 변화가 패턴인식 분야에서는 패러다임의 변화를 일으킬 만큼 대단한 혁신이다. 이러한 혁신을 요약하면 크게 두 가지인데, 하나는 해당 분야의 전문가의 지식 없이도 데이타로부터 자동적으로 필요한 특징들 추출해 낸다는 것이고, 또 하나는 특징 추출과 분류기가 하나의 모델로 통합됨으로써 패턴인식의 성능이 극대화 된다는 점이다.

본 논문에서는 심층망을 이해하기 위한 다양한 개념들, 학습 원리 및 주요 모델들을 설명하고, 이들이 실제 패턴인식 분야에서 어떻게 적용되는



(그림 1) 빅데이타와 이를 효과적으로 활용하는 딥러닝에 기반한 패턴인식 성능의 비약적 발전

지를 사례 중심으로 설명함으로써 딥러닝에 대한 전체적인 이해를 돕고자 한다.

2. 배 경

2.1 신경망과 딥러닝의 역사

앞서 딥러닝은 심층망에서의 학습과 추론에 대한 연구이며, 심층망은 기존 신경망의 계층을 확장한 형태라고 설명하였다. 따라서, 딥러닝을 이해하기 위해서는 신경망의 발전부터 살펴볼 필요가 있다.

논란의 여지는 있겠지만, 최초의 신경망은 '신경망의 아버지' 라고 불려지는 D. Hebb에 의해 1949년 시작되었다고 할 수 있다. 헵은 신경망을 학습하기 위해 헤비안 학습(Hebbian learning) 을 제안했는데, 이것은 한마디로 말하면 같이 행동하는 뉴런들을 더 단단히 연결하라는 학습 원리이다. 단순한 원리이지만, 아직도 많은 경우에 사용되는 방법이다. 이후 1958년 F. Rosenblatt이 단층 신경망인 퍼셉트론(Perceptrons)을 IBM 704에 구현하여 이미지 인식을 수행했다. 이때부터 사람들은 신경망으로 곧 인간 수준의 인공지능을 곧 만들어 낼 수 있을 것이라고 믿기 시작했다. 하지만, 1969년 MIT의 M. Minsky 교수가 단층 신경망은 XOR 문제를 풀 수 없음을 증명함으로써 사람들은 신경망의 능력을 불신하게 되었다[7]. 이때 이미 신경망의 계층을 늘려 계산 능력을 키우려는 생각들이 있었지만, Minsky 교수는 심층망을 만든다 하더라도 신경망은 가능성이 없다고 생각했다.

신경망에 대해 사람들이 다시 열광하기 시작한 것은 1986년 D. Rumelhart, G. Hinton, 그리고 R. Williams이 발표한 역전파(backpropagation) 알고리듬의 등장이었다[8]. 사실 역전파 알고리즘은 그 전에도 있었지만, 1986년 이들의 논문으로부터 다시 주목받기 시작했고, 신경망은 또다시 낙관적인 전망으로 사람들의 관심을 끌어모았다. 이 역전파 알고리즘은 단층 신경망 뿐만 아니라 한 두개의 은닉층을 가지는 다단계(multi-layered) 신경망도 학습가능하게 만들었다. 하지만, 1995년 V. Vapnik 과 C. Cortes에 의해 SVMs (support vector machines) 이 소개되고, 신경망보다 더 좋은 성능을 보이자, 사람들은 다시 신경망을 버리고 SVMs 으로 몰려갔다.

이후 10여년간 신경망은 연구자들의 무관심과 홀대를 받았지만, 토론토대학 Hinton 교수의 2006년 Science 논문을 기준으로 다시 사람들의 주목을 받기 시작했다. 그리고, 패턴인식의 패러다임을 바꾸고, 음성인식, 영상인식등의 분야에서 성공적으로 적용되고 있다. 뿐만 아니라, 언어이해와 같은 분야에서도 성과를 내면서 인공지능의 수준을 한단계 성숙시키는 기술로 인정받고 있다. 버클리대학의 M. Jordan 교수를 비롯한 기계학습의 대가들 중에는 신경망의 성공에 대해 너무 환호하지 말것을 주문하기도 하는데[30], 이는 이미 몇차례 신경망에 대한 기대와 좌절을 경험했기 때문에 신중하자는 뜻으로 해석할 수 있다.

현재 다양한 패턴인식 대회 및 사용 서비스를 위해 가장 많이 사용되는 딥러닝 모델은 CNNs (Convolutional Neural Networks)와 RNNs (Recurrent Neural Networks)이다. 이들은 이미 1980년대에 제안되었고, 많은 논문들 또한 존재한다. 특히, CNNs의 역사는 Hubel 과 Wiesel 의 단순세포(simple cell) 와 복합세포(complex cell) 연구로까지 거슬러 올라간다. CNNs의 첫번째 계산 모델은 Fukushima가 1980년대에 발표한 Neocognitron이다[9]. 이후 1989년 Y. LeCun 이 Neocognitron 에 역전파 알고리즘을 결합하여 CNNs 을 만들었다. CNNs는 미국의 개인 수표에

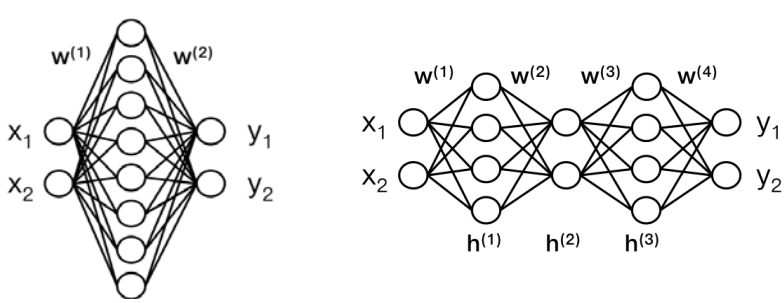
있는 숫자 인식에 사용되는 등, 상업적으로도 큰 성공을 거두었다. 최근 영상인식의 경향은 CNNs 의 규모를 더 확장하고 다양한 구조를 갖게 디자인하여 성능을 극대화 하는 추세다. RNNs 의 경우에는 시계열 데이타 분석을 위한 신경망으로써 최근에는 RNNs 의 일종인 LSTM (long short-term memory) 이 필기체 인식이나 음성인식에 성공적으로 적용되고 있다[10].

2.2 딥러닝의 필요성

신경망이 하나의 은닉 계층만 가지고 있어도, 보편 근사기(universal function approximators) 로 작동한다는 것은 잘 알려진 사실이다. 이 말은 적절한 인공신경망의 구조를 디자인하고 연결강도를 결정하면 어떠한 함수라도 근사적으로 표현할 수 있다는 것이다[12]. 그렇다면 우리는 왜 하나보다 더 많은 여러개의 계층을 쌓는 것이 어떤 이점이 있는지를 생각해 보아야 한다. 주어진 데이타를 표현하거나 입력과 출력간의 관계를 충분히 표현하기 위해서는 그만큼 모델이 복잡해야 하는데, 천층망(shallow networks)에서는 노드의 개수를 증가시는 것이 유일한 방법이다. 이러한 방법은 계층을 쌓아 올리는 것에 비해 효과적이지 못하다.

(그림 2)는 동일한 수의 연결을 가지는 두 가지 신경망 구조의 예를 보여준다. 두개의 신경망은 비슷한 복잡도를 가지고 있다고 할 수 있지만, 심층망은 보다 많은 표현 능력을 가진다고 할 수 있다. 이는 x_1 에서 y_1으로 가는 길의 개수를 세는 것으로 설명할 수 있다. 즉, 천층망에서는 8개의 길이 있고 심층망에는 32개의 길이 있는데, 이는 심층망이 입력과 출력 사이를 더 많은 방법으로 모델링 할 수 있다는 것을 의미한다.

심층망에 대한 생물학적인 연관성도 심층망에 대한 기대를 높인다. SVMs 이나 MLPs 등의 천층망이 많은 패턴인식 문제에 성공적으로 적용되어 왔지만, 음성인식이나 영상인식에서는 여전히 인간 두뇌의 성능에 미치지 못했다. 따라서, 인간의 두뇌에 있는 생물학적 신경망의 원리들을 이해하는 것이 인공신경망의 성능을 향상하는데 도움이 될 것으로 기대해 왔다. 인간두뇌는 영상인식에 있어서 기본적으로 5~10개의 계층을 통해 연산을 수행한다[15]. 즉, 어려운 문제에 대해 심층망의 성공적인 예가 이미 우리 두뇌에서 작동하고 있다는 뜻이다. 또한 MIT 의 T. Poggio 교수나 Google 의 미래학자 R. Kurzweil 은 계층적 모델은 인간수준의 지능을 위해 필수적인 원리라고 주장한다[13,14]. 이는 신경망을 더욱 깊이 만드는 것이 인공신경망을 패턴인식등의 여러가지 지능적인 태스크에 잘 작동할 수 있게 하는데 필수적이라는 뜻이다.



(그림 2) 총 32개의 연결을 가지는 천층망 (왼쪽) 과 심층망 (오른쪽) 구조 예.

2.3 패러다임 변화

한편으로는 심층망이라는 것이 기존의 신경망에 단순히 몇개 계층을 더 증가시킨 것으로 특별할 게 없어 보이기도 하지만, 이러한 양적 증가가 패턴인식에 있어서 패러다임을 변화시킬만큼 큰 변화를 가져왔다. 패러다임의 변화는 크게 두가지로 요약되는데, 첫째는 해당분야의 전문 지식 없이도 데이타로부터 자동으로 특징을 추출해낼 수 있다는 것이고, 둘째는 이것이 기존의 특징 추출기(feature extractor) 와 분류기(classifier) 를 대규모의 신경망으로 통합하여 학습함으로써 독립적인 성능 향상에 비해 성능 개선을 이루었다는 점이다. 이러한 변화는, 예를들어, 의료 영상 분석에서 의사들의 사전 지식에 의존하던 기존의 패턴인식 방법에서 심층망 학습 만으로 단순화되는 것을 의미한다. 아래의 <표 1>에서는 천층망 기반 학습과 심층망 기반 학습의 차이를 요약했다.

〈표 1〉 패턴인식에 있어서 천층학습으로부터 심층학습으로의 패러다임 변화

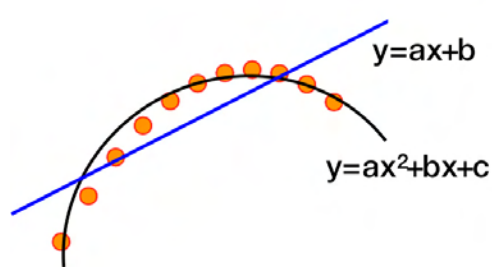
천층 학습	심층 학습
– 분야 전문가에 의한 특징 추출 (예, 음성 MFCC, 동영상 SIFT)	– 데이타로부터 자동 특징 추출
– 특징 추출과 분류기의 독립 개발	– 특징추출과 분류기의 통합

3. 학습 원리

데이터로부터 지능을 얻는, 즉, 무엇인가를 배우는 방법은 크게 감독학습, 무감독학습 및 강화학습으로 나눠진다. 패턴인식과 같은 특정 임무에 관하여 데이터로부터 배운다는 것은 주어진 모델의 변수(parameters)를 조정하여 인식 정확도와 같은 성능의 최대화를 이루어 가는 과정이다. 예를 들어, 아래 (그림 3)에서는, 선형모델의 경우 두 개의 변수 (a,b), 2차식에서는 3개의 변수 (a,b,c)를 조정하여 입력 값 x 에 대해 출력 값 y 를 예측할 수 있게 한다. 그림에서처럼, 3개의 변수를 갖는 모델은 2개의 변수를 갖는 모델을 포함한다. 일반적으로 더 많은 변수를 사용하여 정의된 복잡한 모델일수록 입력과 출력사이의 관계를 더 잘 정의할 수 있다.

이러한 모델 중에 딥러닝은 인공 신경망이라는 특수한 모델에 기반한다. 이름에서 알 수 있듯이, 인공 신경망은 네트워크 구조를 갖는다. 주어진 인공 신경망의 네트워크에 대해 깊이를 정의할 수 있는데, 딥러닝은 기존의 신경망에서 다루는 네트워크들 보다 더 큰 깊이를 갖는 인공 신경망이라고 말할 수 있다. 딥러닝의 최근의 성공은 기존의 신경망보다 더 복잡한 모델을 사용하여 인식 성능을 향상시킨 결과라고도 볼 수 있다.

이번 장에서는 신경망에서의 기본적인 학습 방법과 신경망의 계층을 증가시켜서 발생되는 문제들을 다뤄보고, 심층신경망 혹은 심층망의 성능을 향상시키기 위해 최근 제안되어 많이 사용되는 학습 기법들을 소개한다.



(그림 3) 1,2차식을 사용한 회귀분석 예. 동그란 점들이 (x,y) 로 이루어진 데이터

3.1 신경망에서의 학습

두뇌 신경망의 정보처리 과정에서 고안된 계산 모델로써 인공신경망은 생물학적 신경망에서의 연산단위(혹은 뉴런)와 연결(혹은 시냅스)을 노드(node)와, 그들 사이의 연결(edge), 그리고 매 연결마다 정의되는 연결강도(weight)로 구현한다 (그림 4 참고).

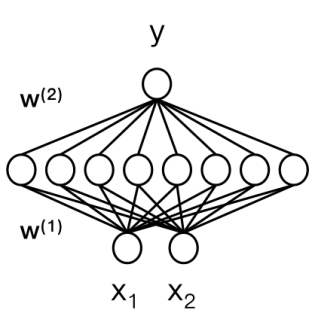
인공신경망에서는 연결강도가 변수이다. 입력값 $x=[x_1,x_2,\cdots x_d]$ 가 주어질 때, 이 값은 계층 h를 거치면서 출력값 y로 변환되는데, 그 과정은 다음과 같이 연결강도 w의 곱과 비선형 함수 σ (주로 sigmoid 함수)의 적용으로 진행된다.

$$h_j = \sigma\left(\sum_i w_{ij}^{(1)}\ x_i\right),$$

$$y = \sigma\left(\sum_j w_j^{(2)} h_j\right).$$

즉, 인공신경망은 연결 강도가 고정되었을 때, 입력값에 대한 출력값을 계산하는 함수로 생각할 수 있다.

특정 패턴인식 임무를 함수라고 가정하면 인공신경망의 학습은 연결강도를 조정하여 이 함수를 찾는 과정이라고 생각할 수 있다. 지도 학습은 학습을 위해 이 목표 함수의 입력값과 출력값의 샘플을 이용하는 방법이다. 학습의 아이디어는 현재의 연결강도로 정의되는 함수에 샘플의 입력값을 대입했을 때의 출력값과 목표 함수에 동일 입력을 대입했을 때의 출력값을 비교한 다음, 이 차이를 감소시키는 새로운 연결강도를 찾는 것이다. 따라서, 출력값 사이의 차이를 표현하는 비용함수와 비용함수를 감소시키는 연결강도 업데이트 방법에 따라 다양한 학습 방법을 고안할 수 있다.



(그림 4) 하나의 은닉계층을 가진 천층망 구조의 예. 원들은 노드, 선들은 연결강도

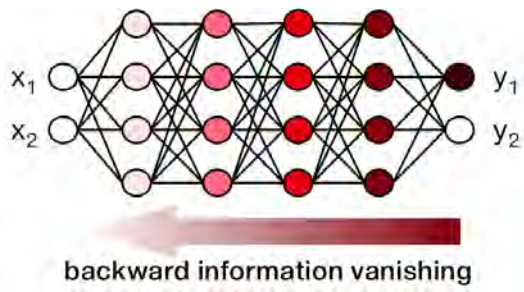
비용함수 E는 주로 출력값 y와 목표값 t사이의 차이, 즉, $E=(y-t)^2$로 정의한다. 학습 과정은 에러의 역전파를 사용하는데, 연결강도의 업데이트는 다음과 같이 정의된다.

$$w_{ij} = w_{ij} - \eta \frac{\partial E}{\partial w_{ij}}\ .$$

여기서 η는 학습률(learning rate)을 의미하며 한번에 어느정도를 학습할지 크기를 결정한다. 각 에러에 대한 연결강도의 편미분은 연쇄법칙(chain rule)을 통해 계산되고 이때 업데이트 수식은 출력값에서부터 시작된 에러의 역전파된 형태로 나타나게 된다. 이런 학습 과정은 매우 단순하고 한 두개의 계층을 포함하는 천층망에서는 잘 작동한다.

3.2 심층망의 어려움

신경망의 계층을 많이 쌓은 심층망이 패턴인식 등의 성능향상에 도움이 된다는 것을 알고 있으면서도 최근까지 심층망이 활발히 연구되지 않은 이유는, 학습이 어렵다는 것이다. 즉, 신경망을 학습하는데 사용되는 역전파 알고리즘이 심층망에서는 에러의 역전파에 어려움을 겪는 것이다. 이러한 역전파의 어려움은 사라지는 경사(vanishing gradient)라는 현상 때문인데, 이는 (그림 5)에서 처럼, 에러 정보가 출력노드에서 입력



(그림 5) 사라지는 경사 현상. 에러 정보는 역전파가 진행될 수록 점차 사라져간다.

노드 방향으로 전달되면서 점점 사라지는 것을 말한다. 에러 정보가 낮은 계층까지 잘 전해지지 않으면서 낮은 계층의 연결강도는 학습 정도가 미미한 수준에 머무르면서 초기의 랜덤 값에서 크게 벗어나지 못하게 된다.

앞서, 인공신경망이 연결강도를 조정하여 다양한 함수를 표현할 수 있다는 점을 언급했다. 하지만, 사라지는 경사 현상으로 인해 상대적으로 낮은 층은 초기의 연결강도 값을 그대로 갖게 되므로, 결국 학습 과정에서 상위 몇 개 층의 연결강도만을 조정하게 된다. 이는 결국 상대적으로 적은 깊이를 갖는 모델을 사용한 것과 같기 때문에 성능 정확도를 향상시키는데 실패할 수 밖에 없다.

하지만, 2006년 G. Hinton 교수가 사전학습(pretraining)을 제시함으로써 심층망의 학습 가능성을 보여줬고, 이후 다양한 방법들이 제안되고 있다. 다음 섹션에서는 심층망 학습에서 중요한 기법들을 소개한다.

3.3 심층망을 위한 학습 기법

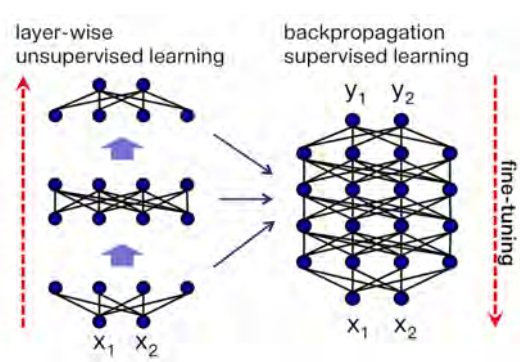
심층망의 학습을 가능하게 하는 방법들 중에 모델에 상관없이 보편적으로 사용될 수 있는 대표적인 방법들 몇가지를 살펴보자.

3.3.1 사전학습(Pretraining)

사전학습은 매우 간단한 아이디어로써, 심층망에 역전파 알고리즘을 적용하기 전에 각 계층별로 사전학습을 진행하는 것이다. 즉, 역전파 알고리즘을 임의의 값(random) 에서 시작하는 것이 아니라, 사전학습을 통해 심층망의 연결을 학습에 도움이 되는 중간 값으로 미리 변형해 놓는 것을 의미한다. (그림 6)에서 보이는 것 처럼, 입력값이 주어지면, 첫번째 계층을 먼저 학습하고, 그 출력값을 두번째 계층의 입력으로 사용하여 두번째 계층을 학습한다. 이러한 과정을 모든 계층에 순서대로 진행한다. 즉, 전체 신경망을 층별로 분해해서 학습하는 것이다. 이후 역전파 알고리즘으로 전체 신경망을 학습하는데 이를 미세조정(fine-tuning) 이라고 한다. 이름 그대로 미세조정을 통해서는 연결강도가 아주 조금 조정된다.

이런 사전학습은 초기값을 최적해 근처로 옮겨 놓는다는 점에서 최적화(optimization) 문제를 위한 좋은 초기해를 찾는 방법으로 해석할 수 있다. 뿐만 아니라, 무감독학습이 $p(x)$로 표현되는 데이타의 분포를 학습하고, 감독학습에 기반한 미세조정은 $p(y|x)$ 로 표현되는 분류성능을 최대화 하는데, 베이즈 룰(Bayes rule)에 따라, 좋은 $p(x)$는 $p(y|x)$, 즉 분류 문제에 대한 좋은 사전 지식이 된다.

사전학습의 또 다른 장점은 무감독학습이기 때

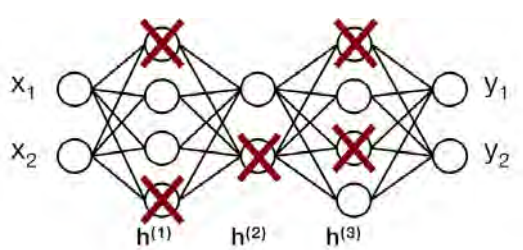


(그림 6) 심층망 학습을 위한 사전학습.

문에 레이블 없는 빅데이타를 학습에 사용할 수 있다는 점이다. 감독학습을 위해 필요한 레이블이 많지 않은 데이타들도 있고, 또 레이블을 만드는데 드는 비용이 매우 큰 경우 무감독 학습은 유용하다.

3.3.2 Dropout

Dropout 은 (그림 7)에서 처럼 학습하는 중에 노드들의 절반 (꼭 절반일 필요는 없다) 을 임의로 끄고 진행한다. 매 학습 회수마다 임의의 선택을 새로 한다. 학습이 끝난 후 새로운 데이타에 대해서는 절반의 노드를 끄는 대신 모든 노드들의 출력값을 절반으로 나눈다. 이러한 방법은 기계학습의 bagging 방법과 비슷한 효과를 만드는데, 안정성과 정확도를 향상시킨다[11]. 그리고 중요한 것은 dropout 은 상호적응(coadaptation) 문제를 해소한다. 두개의 노드가 한번 비슷한 연결강도를 가지게 되면, 그 두 노드는 비슷한 방식으로 업데이트 되면서 마치 하나의 노드처럼 작동하고, 이것은 컴퓨팅 파워와 메모리의 낭비이다. Dropout이 임의로 노드들을 끌때 이러한 두개의 노드가 나눠지게되면 상호 적응 문제를 회피할 수 있게 된다.

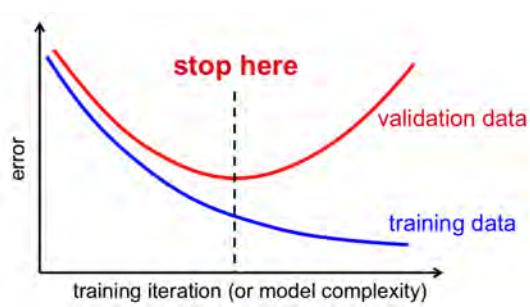


(그림 7) Dropout 예. 학습중에 매번 전체 노드의 임의의 절반을 지우고 학습한다.

3.3.3 조기 멈춤 (Early stopping)

심층망과 같은 패턴인식을 위한 모델을 학습할 때는 보통 두 가지 목표를 동시에 달성하기를 원한다. 하나는 비용함수를 최소화하는 모델을 찾는 것이고, 다른 하나는 찾은 모델이 학습에 사용되지 않은 데이터에 대해서도 인식을 잘하기를 원하는 것, 즉 과적합(overfitting)을 피하는 것이다. 과적합 문제는 근본적으로 학습 데이터에 대해 비용함수를 최소화하기 때문에 발생한다. 즉 모델이 지나치게 학습 데이터에만 적합하게 만들어져서 학습에서 보지 못한 데이터에 대해서는 큰 에러를 주는 경우이다. 이를 해결하기 위해서 주로 쓰는 방법은 조기 멈춤(early stopping)이다 (그림 8 참고). 이 방법은 학습 데이타 중 일부를 검증데이타로 따로 떼어놓고, 남은 데이타로만 학습을 진행한다. 학습 중 검증 데이타로 성능을 검증해서 에러가 떨어지다가 올라가기 시작하면 학습을 멈춘다. 이 방법은 매우 간단해 보이지만, 잘 작동한다.

그외에도 maxout 이나 선형 정류기(linear rectifier) 등의 모델 관련 기법들과, GPU (graphic processing unit) 와 여러대의 서버들에 기반한 병렬연산(parallel computing) 기법, 심층망의 비효율적인 학습결과들을 극복하기위해 천층망으로 심층망의 성능을 모방하는 모델 압축(model compression) 기법들이 주요하게 사용되거나 최



(그림 8) 학습을 계속 진행하지 않고 검증데이타의 에러가 증가하기 시작할때 멈춘다.

근 연구되고 있지만, 본 논문에서는 지면 관계로 생략한다.

3.4 심층망의 장점

심층망을 통한 장점들은 여러가지로 개념으로 설명될 수 있다. 즉, 분산표현(distributed representation), 계층화(hierarchy), 추상화(abstraction), 재사용성(reusability), 다양체(manifold), 풀기(disentanglement), 지식전달(knowledge transfer) 등등이 있는데 딥러닝을 이해하는데 필수적인 계층화와 풀기를 중심으로 간단히 설명한다.

심층망에서 계층화는 재사용성이나 다양체와도 깊이 관련된다. 하지만 추상화에 더 직접적으로 관련을 가지는데, 계층이 증가될수록 더 추상적인 표현을 찾게 된다는 의미이다. 입력되는 데이타 (얼굴 인식의 경우 얼굴 이미지) 는 가공되지 않은 그대로 이고, 출력되는 데이타는 가장 추상적인 표현 (얼굴 인식의 경우 얼굴 ID) 일때, 계층의 수가 많아질 수록 그 추상화의 단계는 세분화되고, 상위층으로 올라갈 수록 추상화의 정도가 점진적으로 높아진다. 계층을 올라갈 수록 비선형 함수에 의해 아래 계층의 표현을 조금 더 출력값을 닮아 가는 방향으로 표현은 변화되기 때문이다. 이러한 추상적인 표현은 데이타에서 일어나는 작은 변화들에 강건한 특징 (invariance to local variations) 을 갖게 된다. 이러한 추상화 과정은 출력이 정수와 같은 이산적인 클러스터링 (clustering) 이나 실수와 같은 회귀(regression) 모두에 동일한데, 입력값중에 출력값에 관련된 표현들만 찾아내는 과정으로 이해할 수 있다.

딥러닝을 가능하게했던 사전학습이 분류 문제에 있어서 도움이 되는 이유를 설명하는 다양한 방법이 있겠지만, 설득력있는 설명이 바로 풀기이다. 분류의 경우 주어진 데이터에 대한 클래스를 찾아내는건데, 각 클래스를 설명하는 요소들은 데이타 곳곳에 녹아있다고 볼 수 있다. 사전학습의 경우 섞여있는 요소들, 즉 클래스를 설명하는 요소들을 풀어내어 클래스를 찾기 쉽게 해주는 것이다. 사전학습에서는 어떤 요소가 분류문제에 도움이 되는지 알 수 없기 때문에 가능한 모든 요소들을 풀어내는 것이 중요하다. 다양체는 데이타의 변화 중 중요한 부분들만 표현하는 것이라면, 풀기는 모든 요소들을 추출하여 표현하는 것이 목표다. 미세조정의 경우에도 하나의 임무만 주어진 경우라면 노이즈에 해당하는 부분들을 버리는 다양체가 도움이 되겠지만, 다음 임무를 위해 지식전달이 필요한 경우에는, 어느 것이 노이즈 인지 알 수 없고, 따라서 모든 요소들을 다 추출하여 표현하는 것이 도움이 된다.

4. 주요 알고리즘

딥러닝에는 다양한 모델과 알고리즘들이 있다. 여기서는 그러한 알고리즘들을, 심층망을 이루는 구성요소들, 심층망중에는 생성모델과 판별모델 등의 세 그룹으로 나누고 간략히 설명한다.

4.1 구성 요소

심층망을 구성하는 구성요소들은 여러가지가 있는데, 주로 RBMs (restricted Boltzmann machines) 이나 AEs (auto-encoders) 가 사용된다. 두 모델의 구조는 아래 (그림 9)와 같다.

RBMs 은 1986년에 소개된 생성모델이고 최근 딥러닝에 구성요소로 사용되면서 주목을 끌게 되었다[18]. (그림 9)의 네트웍 구조에서 관측(visible) 노드들은 은닉(hidden) 노드들과만 연결되어있고, 학습은 아래의 확률을 최대화 하는 방향으로 이루어진다. 학습과정은 경사하강법(gradient

(그림 9) RBM 구조 (왼쪽) 와 AE 구조 (오른쪽).

descent method) 를 사용할 수 있지만, 주로 CD (contrastive divergence) 라는 방법을 사용한다[6].

$$p(v,h) = \frac{1}{Z} e^{-E(v,h)},$$

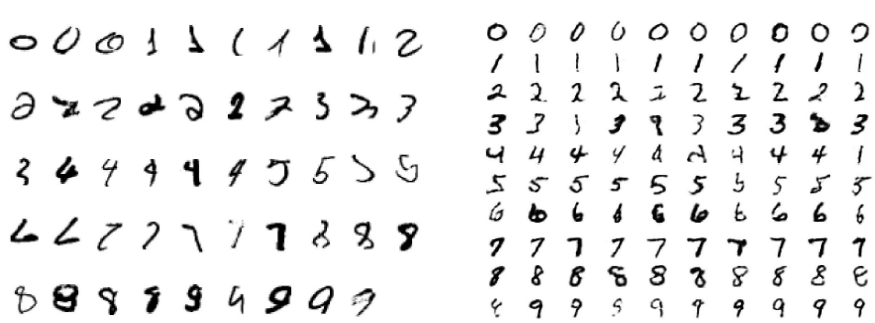
$$E(v,h) = -v^T W h.$$

AEs는 (그림 9) (오른쪽)에서처럼 2 계층 신경망인데, 입력과 출력은 동일하다. 인코딩 행렬 W에 대해 디코딩은 W의 전치 (transpose)로 표현할 수 있어서 동일한 변수를 사용한다. 이때, 학습은 복구에러의 역전파에 기반한다. 최근에는 제한된 (restricted) AEs 가 제안되어 다양하나 형태로 사용되고 있다[17].

4.2 생성모델

딥러닝 연구 성과의 상당부분은 감독학습에 기반한 판별모델로부터 나온 결과들이지만, 무감독 학습의 생성모델도 딥러닝 연구의 큰 축을 이루고 있다. 뿐만아니라, 생성모델은 인공지능에 있어서도 굉장히 중요한데, 모델이 데이타를 생성해낼 수 있다는 말은 그 모델이 그 데이타를 잘 이해했다고 판단할 수 있는 근거가 되기 때문이다. 본 논문에서는 심층망중 가장 유명한 생성모델인 DBNs (deep belief networks) 를 간략히 살펴본다.

DBNs 은 신경망과 같은 네트워크 구조인데 최상위 층은 RBM 형태의 무향 그래프(undirected graph), 그 아래 계층들은 모두 top-down 구조의 유향 그래프로 구성된다. DBNs의 학습은 먼저, 사전학습된 RBMs 을 쌓은 후, 제일 위층은 RBM 그대로 두고 나머지는 유향 그래프(directed graph)로 두고 up-down 알고리즘을 사용하여 미세조정 하는 방법을 따른다[6]. 학습 후 DBNs 는 학습에 사용된 데이타와 같은 종류의 데이타를 생성해낼 수 있는데, 두가지 방법을 통해서 생성한다. 첫번째는 제일 위층에서 노이즈를 생성한



(그림 10) MNIST 데이타와 (왼쪽) 생성된 데이타 (오른쪽)[6].

다음, RBM 내에서 반복적인 샘플링을 통해 어느 정도 개념이 생성되면 아래로 내려오는 방법, 둘째는 제일 아래에 실제 데이타를 넣고 (혹은 노이즈를 넣고) 제일 위층까지 올라간 다음 다시 내려오는 방법이 있다. 즉 최상위 계층이 데이타의 개념을 표현한다. (그림 10)은 MNIST 데이타로부터 DBNs 을 학습한 후 새로 생성한 데이타를 보여주는데, 생성한 데이타와 원래 학습 데이타간에 구분이 쉽지 않을 정도로 DBNs 은 데이타를 잘 학습한 것으로 볼 수 있다.

4.3 판별모델

신경망의 부흥을 이끈 것은 여러가지 패턴인식에서 기존 최고 기록들을 경신해 온 판별모델이라고 할 수 있다. Hinton 교수의 초기 모델은 DBN-DNNs, 즉, RBMs 으로 초기화한뒤 역전파로 미세조정하는 모델이였는데, 현재 영상인식에서 가장 많이 사용되는 모델은 CNNs 이고[2,19], 음성인식등의 시계열 데이타인식에는 RNNs 이 가장 많이 사용된다. CNNs 과 RNNs 은 1980년대부터 사용되던 모델로써, 사전학습과 같은 기법들을 사용하진 않고, 바로 감독학습을 수행한다.

CNNs의 중요한 특징중에 하나는 뇌신경과학적인 발견들에 기초한 모델이라는 점이다. Hubel-Wiesel의 단수-복잡 세포라던가, 지역적 감각수용장(local receptive fields), 뽑기(pooling) 같은 것들은 뇌과학으로부터 나온 개념들이다. CNNs 은 이러한 개념들위에 지역적 감각수용장을 전체 이미지에 합성곱 (convolution) 하여 신경망의 연결강도를 공유하는 방법으로 변수의 개수를 대폭 축소한다. CNNs 이 사전학습 없이도 학습 가능한 이유는 이러한 지역적 연결과 공유된 연결이 역전파되는 에러정보를 사라지지 않게 하고 에러의 분산을 막기 때문이다.

CNNs 이 영상인식에 최적화된 구조를 가지고 있는 반면, 시계열 데이타에 대해서는 RNNs 이 적절한 구조를 가지고 있다. RNNs 은 일반 신경망의 각 계층에서 상위 계층으로 연결 뿐만아니라 자기 자신 계층에도 연결을 만드는데, 이러한 돌아오는 연결은 마치 메모리와 같은 역할을 함으로써 데이타의 시간적인 변화를 모델링할 수 있게 만든다. 반면, 이러한 돌아오는 연결로 학습은 더욱 어려워진다[20]. 최근에는 긴 단기기억 모델(LSTM) 등을 사용하면서 이러한 학습 문제들이 해소되었고, 이를 사용한 RNNs 이 음성인식 분야에서 우수한 성능을 내고 있다[10].

4.4 소스코드

딥러닝에는 다양한 알고리즘이 있고 이용가능한 다양한 버전들이 구현되어있다. 아래 <표 2>는 이러한 코드들을 요약해두었다. Science2006 과 Cuda-Convnet 은 최초라는 상징적인 의미가

〈표 2〉 딥러닝 알고리즘들의 코드

Name	Codes	Algorithms	Language
Science2006	http://www.cs.toronto.edu/~hinton/MatlabForSciencePaper.html	DBNs, DAEs	Matlab
Cuda-Convnet	https://code.google.com/p/cuda-convnet/	CNNs	C++
Caffe	http://caffe.berkeleyvision.org/	CNNs	C++
Pylearn2	https://github.com/lisa-lab/pylearn2	DBMs, CNNs, GSNs, etc	Python
RNNLIB	http://sourceforge.net/projects/rnnl/	RNNs	C++
CURRENNT	http://sourceforge.net/projects/currennt/	RNNs	C++

있고 Pylearn2는 연구목적에 잘 맞으며 Caffe[21] 나 CURRENNT 가 가장 최근의 코드들로써 사용하기에 좋을 것으로 보인다.

5. 응 용

딥러닝을 이용한 패턴인식의 사례는 매우 많이 있지만, 여기서는 대표적인 몇가지만 살펴본다.

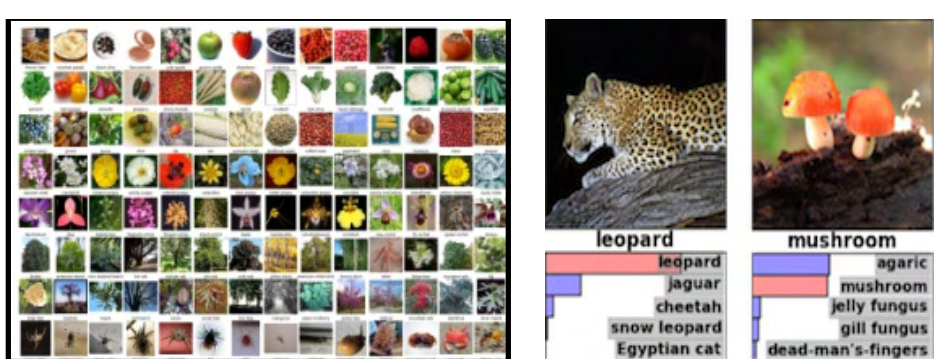
가장 대표적인 예는 2012년 ImageNet 데이타에서의 이미지 인식 이였는데, 이미지가 주어지면 1000개의 클래스 중에 이미지에 가장 맞는 클래스를 맞추는 임무였고 (그림 11)은 그 몇몇 이미지들과 인식 결과 예를 보여준다. 기존의 최고 성능은 21.9% 였지만 CNNs 은 그 성능을 32.6%로 대폭 향상시켰다[2]. 기존의 수십년동안 진행된 컴퓨터 비전 기술들의 성능을 단순한 CNNs 적용만으로 획기적으로 뛰어넘은 것으로 기념비적인 사례이다.

음성인식에 관하여는 2000년 이후 별다른 성능개선이 없다가 2010년 전후로 딥러닝에 의해 대폭적인 개선이 있었다 (표 3 참고). 하지만, 이미지 인식에서 CNNs은 이미지로부터 클래스 정보까지 처리를 하지만 음성인식에서는 아직 전체 시스템의 일부 (음향모델) 만을 딥러닝이 담당하고 있다[31]. 초기에는 DBN-DNNs 방식이 주로 사용되다가 최근에는 LSTM 기반 RNNs 방식이 주목받고 있다.

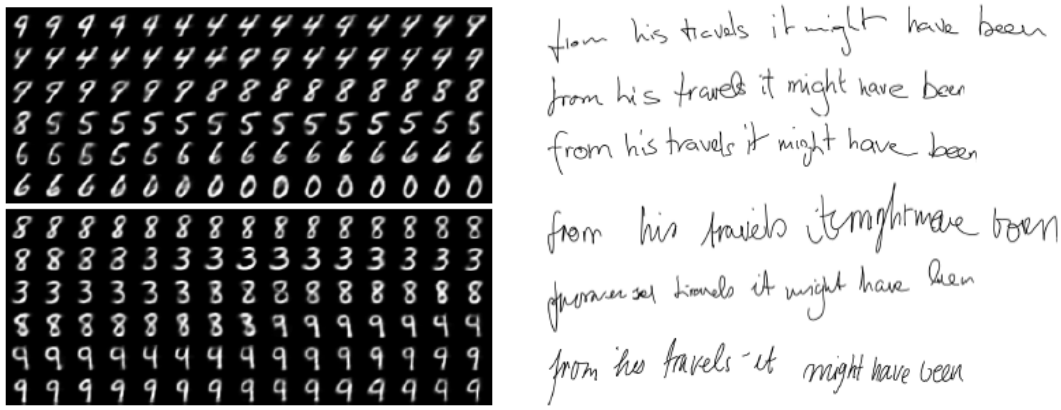
〈표 3〉 음성인식에서의 성능 향상. TIMIT 데이타에서 음소인식 결과[16].

Rank	Method	Accuracy (%)
1	Deep RNN	82.3
2	DBN-DNN	79.3
3	Augmented CRF	77
4	MLP-HMM	74.2

다른 기계학습 알고리즘들과 달리, 심층망에 기반한 딥러닝은 이미지나 음성 인식 등의 패턴인식 뿐만 아니라 다양한 문제들에 잘 적용될 수 있는데, 언어이해(language understanding), 다양한 종류의 데이타 처리(multimodal learning), 지식 전달(knowledge transfer or transfer learning), 데이타 생성(data generation) 등이 그러한 예가 된다[22-25]. 아래 그림 5.2는 GSNs (generative stochastic networks) 과 RNNs 에 의해 생성된 데이타 예들이다. 왼쪽은 필기체 숫자 데이타로 부터 학습된 GSNs 이 필기체 숫자 이미지를 생생해내는 과정이고, 오른쪽은 필기체 문장을 배운 RNNs 이, 주어진 문장을 필기체로 쓴 결과이다.



(그림 11) ImageNet 샘플 이미지 (왼쪽) 과 인식결과 예 (오른쪽).



(그림 12) GSNs 가 생성한 필기체 숫자 (왼쪽) 와 RNNs 의 필기체 문장 (오른쪽)[24,25].

6. 결 론

2006년 이후 딥러닝은 매우 많은 개선과 새로운 시도들을 포함해왔는데, 이러한 성과에는 컴퓨팅 파워와 빅데이처리에 관한 기술들이 바탕이 되었다. 딥러닝은 지금도 다양한 연구들이 진행되고 있는데, 이러한 연구들 중에 주요한 방향은 크게 몇가지 형태로 요약될 수 있다. 하나는 대규모 모델을 빠르고 효과적으로 학습하는 방법에 관한 최적화 연구이고, 또 하나는 다양한 사례들에 맞게 신경망 구조를 변형하고 적용해서 성능을 개선하는 연구이다 [29,17]. 특히나 여러개의 GPU를 활용하여 속도를 개선하는 방법이나, 학습 알고리즘 자체를 개선하는 연구들이 활발히 진행되고 있고, 번역이나 이미지로부터 자막 자동생성과 같은 기존에는 불가능해보였던 영역들에서 딥러닝이 성과를 내고있다.

딥러닝은 패턴인식 분야 뿐만아니라 더 넓게는 인공지능 분야에 있어서 큰 획을 그었다고 할 수 있다. 하지만, 이러한 개선도 D. Marr 에 의해 주장된 수동적 정보처리의 한계를 뛰어넘지는 못하고 있는데, 인공지능 분야에서 더 많은 발전을 이루기 위해서는 환경과 상호 작용하는 로봇이 필요하다는 R. Brooks 교수의 주장에 귀 기울일 필요가 있다 [26,27]. 최근 강화학습과 딥러닝의 결합을 위한 시도는 이러한 방향을 따르는 것이다.

신경망 학습 알고리즘들이 개선되고 있을 뿐만 아니라, 뇌신경과학적 발견들과 빅데이타 처리에 대한 노하우가 쌓여가고, GPU 의 발전을 포함한 컴퓨팅파워가 증가함으로써 딥러닝의 발전도 더욱 가속화 될 것으로 전망된다. 이러한 시점에서 딥러닝의 배경을 살펴보고 주요 이슈들을 점검하는 것은 딥러닝 연구를 시작하는 이들에게 유용할 것으로 기대된다.

참 고 문 헌

[1] G. Hinton, R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, Vol. 313, No. 5786, pp. 504–507, Jul. 2006.

[2] A. Krizhevsky, I. Sutskever, G. Hinton, “ImageNet classification with deep convolutional neural networks,” Advances in Neural Information Processing (NIPS), Lake Taho, NV, 2012.

[3] J. Markoff, “How Many Computers to Identify a Cat? 16,000,” *New York Times*, June 25,

2012.
[4] J. Markoff, “Scientists See Promise in Deep-Learning Programs,” *New York Times*. November 24, 2012.
[5] G. Marcus, “Is ‘Deep Learning’ a Revolution in Artificial Intelligence?” *The New Yorker*, November 25, 2012.
[6] G. Hinton, S. Osindero, Y. Teh, “A fast learning algorithm for deep belief nets,” *Neural Computation* Vol.18, pp. 1527-1554, 2006.
[7] M. Minsky, S. Papert, *Perceptrons,* Cambridge, MA: MIT Press, 1969.
[8] D. E. Rumelhart, G. E. Hinton, R. J. Williams, “Learning internal representations by error propagation” in *Parallel Distributed Processing*, MIT Press, 1986, pp. 318–362.
[9] K. Fukushima, “Neocognitron: A self-organizing neural network for a mechanism of pattern recognition unaffected by shift in position,” *Biological Cybernetics*, Vol. 36, No. 4, pp.193–202, 1980.
[10] A. Graves, J. Schmidhuber, “Framewise phoneme classification with bidirectional LSTM and other neural network architectures,” *Neural Networks*, Vol. 18, No. 5-6, pp.602–610, 2005.
[11] P. Baldi, P. J. Sadowski, “Understanding dropout,” Advances in Neural Information Processing Systems (NIPS), 2013, pp.2814-2822.
[12] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford: Oxford University Press, 1995.
[13] M. Riesenhuber, T. Poggio, “Hierarchical models of object recognition in cortex,” *Nature Neuroscience*, Vol. 2, No. 11, pp.1019–1025, 1999.
[14] R. Kurzweil, *How to Create a Mind: The Secret of Human Thought Revealed*. Penguin Books, 2012.
[15] S. J. Thorpe, M. Fabre-Thorpe, “Seeking Categories in the Brain,” *Science*. Vol. 291, No. 5502, pp.260-262, Jan. 2001.
[16] A. Graves, A. Mohamed, G. Hinton. “Speech Recognition with Deep Recurrent Neural Networks,” International Conf. on Acoustics, Speech, and Signal Processing (ICASSP), 2013, Vancouver, Canada.
[17] Y. Bengio, A. Courville, P. Vincent, “Representation learning: A review and new perspectives,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 35, No. 8, pp.1798–1828, 2013.
[18] P. Smolensky, “Information Processing in Dynamical Systems: Foundations of Harmony Theory,” in *Parallel distributed processing: Explorations in the microstructure of cognition*, *Vol. 1*, MIT Press, Cambridge, MA, 1986, pp. 194–281.
[19] D. C. Ciresan, U. Meier, J. Masci, J. Schmidhuber, “A committee of neural networks for traffic sign classification,” In Proc. of International Joint Conference on Neural Networks (IJCNN), 2011, pp.1918–1921.
[20] Y. Bengio, P. Simard, P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Transactions on Neural Networks*, Vol. 5, No. 2, pp.157–166, 1994.
[21] Y. Jia, “Caffe: An Open Source Convolutional Architecture for Fast Feature Embedding,” 2013, http://caffe.berkeleyvision.org/.
[22] T. Mikolov, W.-T. Yih, G. Zweig, “Linguistic Regularities in Continuous Space Word Representations,” In Proc. of NAACL HLT, 2013.
[23] R. Socher, M. Ganjoo, H. Sridhar, O. Bastani, C. D. Manning, A. Y. Ng, “Zero-shot learning

through cross-modal transfer," In Proc. of International Conference on Learning Representations (ICLR), Scottsdale, AZ, 2013.

[24] Y. Bengio, É. Thibodeau-Laufer, G. Alain, J. Yosinski, "Deep Generative Stochastic Networks Trainable by Backprop," In Proc. of International Conference on Machine Learning (ICML), 2014.

[25] A. Graves. "Generating Sequences With Recurrent Neural Networks," 2014.

[26] D. Marr, "Vision: A Computational Investigation into Human Representation and Processing of Visual Information," Freeman, San Francisco, 1982.

[27] R. A. Brooks, "Elephants Don't Play Chess," *Robotics and Autonomous Systems,* Vol. 6, pp.3-15, 1990.

[28] J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," Technical Report IDSIA-03-14, 2014.

[29] C. Farabet, B. Martini, B. Corda, P. Akselrod, E. Culurciello, Y. LeCun, "NeuFlow: A Runtime Reconfigurable Dataflow Processor for Vision", in Proc. of the Fifth IEEE Workshop on Embedded Computer Vision (ECV), Colorado Springs, 2011.

[30] http://spectrum.ieee.org/robotics/artificial-intelligence/machinelearning-maestro-michael-jordan-on-the-delusions-of-big-data-and-other-huge-engineering-efforts

[31] L. Deng, "Three classes of deep learning architectures and their applications: a tutorial survey," *APSIPA Transactions on Signal and Information Processing*, 2012.

저 자 약 력

최 희 열

이메일 : heeyoul@gmail.com

- 2002년 포항공과대학교 컴퓨터공학과 (학사)
- 2005년 포항공과대학교 컴퓨터공학과 (석사)
- 2010년 Texas A&M University, Computer Science and Engineering (박사)
- 2010년~2011년 Indiana University, Cognitive Science Program (Post-Doc)
- 2011년~현재 삼성전자 종합기술원 전문연구원
- 관심분야 : Deep Learning, Manifold Learning, Cognitive Science, Computational Neuroscience

민 윤 홍

이메일 : yunhong.min@gmail.com

- 2006년 포항공과대학교 산업경영공학과 (학사)
- 2012년 서울대학교 산업공학과 (박사)
- 2012년~현재 삼성전자 종합기술원 전문연구원
- 관심분야 : Deep Learning, Convex Optimization