

# 기계학습 및 딥러닝 기술동향

문성은, 장수범, 이정혁, 이종석  
연세대학교

## 요약

본 논문에서는 패턴 인식 및 회귀 문제를 풀기 위해 쓰이는 기계학습에 대한 전반적인 이론과 설계방법에 대해 알아본다. 대표적인 기계학습 방법인 신경회로망과 기저벡터머신 등에 대해 소개하고 이러한 기계학습 모델을 선택하고 구축하는 데에 있어 고려해야 하는 문제점들에 대해 이야기 한다. 그리고 특징 추출 과정이 기계학습 모델의 성능에 어떻게 영향을 미치는지, 일반적으로 특징 추출을 위해 어떤 방법들이 사용되는 지에 대해 알아본다. 또한, 최근 새로운 패러다임으로 대두되고 있는 딥러닝에 대해 소개한다. 자가인코더, 제한볼츠만기계, 컨볼루션신경회로망, 회귀신경회로망과 같이 딥러닝 기술이 적용된 대표적인 신경망 구조에 대해 설명하고 기존의 기계학습 모델과 비교하여 딥러닝이 가지고 있는 특징점을 알아본다.

## I. 서론

기계학습 전반에 대해 설명하기 위해 먼저 예시를 하나 들어보자. 어떤 가게에서 자동으로 오렌지와 자몽을 구분하는 시스템을 만들고자 한다고 가정할 때, 가장 먼저 해야 할 일은 이미 오렌지 혹은 자몽이라는 사실을 알고 있는 과일들을 관찰하고 어떤 기준으로 그 둘을 구분할 수 있는지 알아내는 것이다. 가게주인은 관찰을 통해 일반적으로 오렌지가 더 노란 빛을 띠고 자몽은 더 붉은 빛을 띤다는 사실을 알아냈다. 하지만 <그림 1>에서 알 수 있듯이 색만으로는 오렌지와 자몽을 만족스럽게 구분해 낼 수 없었다. 가게주인은 다시 관찰을 통해 일반적으로 오렌지가 크기가 더 작다는 사실을 알아내고, 이것을 기준으로 가지고 있는 과일들을 나누어 보았다. 크기로 오렌지와 자몽을 구분한 결과<그림 2>, 색을 사용했을 때보다 만족스러운 결과를 얻을 수 있었다. 그러면 색과 크기를 같이 사용하면 어떨까? 두 개의 값을 같이 사용하니 <그림 3>과 같이 보다 나은 결과를 얻을 수 있었다. 이제 이 가게주인은 새로운 과일이 들어왔을

때 과일의 색과 크기를 측정해 이 그래프에 대입해봄으로써 그 과일이 오렌지인지 혹은 자몽인지 알아낼 수 있게 되었다.

위의 일련의 과정들이 학습데이터(이미 종류를 알고 있는 과일)로부터 특징(feature: 색, 크기)을 추출하고 학습(training/

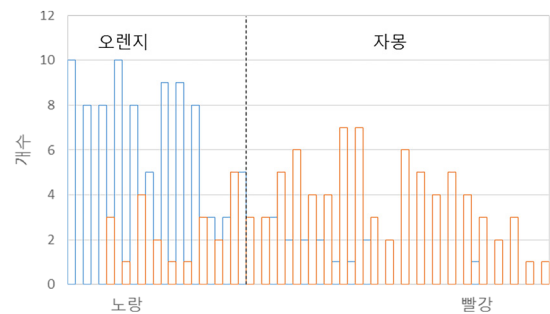


그림 1. 오렌지와 자몽을 색으로 구분한 경우

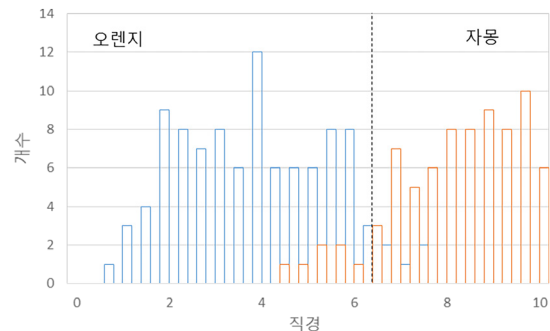


그림 2. 오렌지와 자몽을 직경으로 구분한 경우

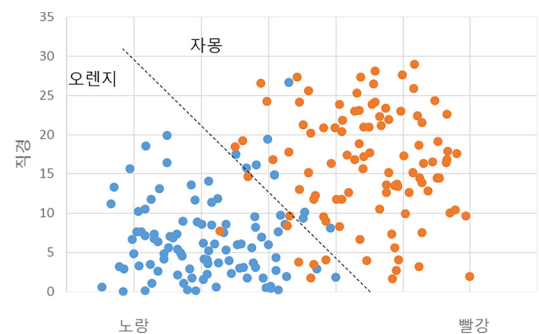


그림 3. 오렌지와 자몽을 색과 직경을 이용해 구분한 경우

learning)을 통해 클래스(오렌지, 자몽)를 구분하는 모델(결정 경계, 그래프의 점선)을 찾은 뒤, 새로운 데이터(새로 들어온 과일)의 특징 값을 기반으로 그 데이터의 클래스를 결정(test/generalization)하는 기계학습의 일반적인 순서를 나타내고 있다.

그런데 그래프를 다시 보면 결정경계(decision boundary)가 오렌지와 자몽을 완벽하게 구분하지 못해 잘못 분류된 데이터가 있다. 그러면 결정경계를 직선으로 굿지 말고 이 데이터들까지 완벽하게 구분할 수 있도록 그으면 좀 더 좋은 시스템이 되지 않을까? 바로 여기에서 기계학습의 과도학습(overfitting) 문제가 발생한다. 일반적인 데이터에는 노이즈가 섞여있기 때문에 주어진 학습데이터에 맞춰 시스템을 과도하게 학습시키는 것은 오히려 시스템의 성능을 저하시킨다. 따라서, 기계학습 시스템의 복잡도는 풀고자 하는 문제의 복잡도와 일치해야 한다.

기계학습을 통해 풀 수 있는 문제는 크게 주어진 데이터의 클래스를 구분해야 하는 패턴인식(pattern classification) 문제와 연속적인 어떤 값을 추정해야 하는 회귀(regression) 또는 함수 근사화(function approximation) 문제로 나눌 수 있다. 또한 학습 방법에 따라서는 패턴인식이나 회귀 문제에서 학습데이터의 클래스나 출력 값을 알고 이에 대한 피드백을 통해 학습하는 지도학습(supervised learning), 학습 데이터의 클래스나 출력 값을 이용하지 않고 데이터 패턴이나 클러스터, 밀도 등을 추정하는 비지도학습(unsupervised learning), 클래스나 출력 값을 아는 데이터(labeled data)와 모르는 데이터(unlabeled data)를 함께 사용하는 반지도학습(semi-supervised learning), 결과에 대한 피드백만 주어지고 정확한 클래스나 출력 값은 주어지지 않는 강화학습(reinforcement learning)으로 나눌 수 있다.

기계학습은 사람이 풀 수 있지만 학습하기에 너무 문제의 규모가 크거나 또는 수학적으로 문제를 정의할 수 있지만 문제가 복잡하여 수학적으로 깨끗하게 정의를 하는 것이 어려울 때에 유용하게 쓰일 수 있다. 기존의 기계학습은 그 성능이 사람의 수준에 미치지 못하거나 하드웨어의 한계로 학습에 오랜 시간

이 소요되는 등의 문제가 있었지만, 최근 딥러닝의 개발, 빅데이터의 활성화, 하드웨어의 발전에 힘입어 처리 가능한 문제의 범위가 급속도로 넓어지면서 여러 가지 분야에 접목되고 있는 중이다. 특히, 예전에는 기계학습이 많이 사용되지 않았던 자연 과학 분야 등에서도 기계학습을 도입하려는 움직임이 나타나면서 기계학습은 인공지능이나 패턴 인식 등에 쓰이는 것을 넘어서 매우 범용적인 기술이 되었다. 본 논문에서는 기존의 기계학습 모델들과 그 한계, 그리고 딥러닝의 개념과 그 가능성에 대해 간략하게 다루며 최근 기계학습이 이루어낸 발전에 대해 살펴본다.

## II. 기계학습 모델

### 1. 신경회로망

신경회로망(neural network)은 사람의 뇌의 구조를 모방한 기계학습 방법으로, 계량적 접근을 통해 결정경계를 찾는다. 사람의 뇌에는  $10^{11}$ 개의 뉴런이 있고 이 뉴런들 사이에  $10^{15}$ 개의 시냅스 연결이 존재한다. 1943년에 McCulloch와 Pitts에 의해 제안된 McCulloch-Pitts neuron은 가중치가 곱해진 입력 값들의 합을 계산하여 그 합이 임계값을 넘으면 1, 아니면 0을 출력하는 인공뉴런이다[1]. 1957년 Rosenblatt는 이 인공뉴런의 개념을 기반으로 퍼셉트론(perceptron)이라고 불리는 뉴런의 입력 값에 곱해지는 가중치를 학습하는 인공신경망 모델을 만들었다[2]. 퍼셉트론은 다양한 문제에 적용할 수 있을 것으로 기대되며 많은 주목을 끌었지만, Minsky와 Papert는 1969년에 출판한 저서에서 퍼셉트론이 XOR과 같이 선형으로 분리되지 않는 문제는 해결할 수 없다는 사실을 지적한다[3].

이러한 문제는 다층 신경회로망(multi-layer neural network)을 사용함으로써 풀리게 된다(그림 4). 그러나 다층 구조를 사용하게 되면서 몇 개의 은닉층(hidden layer)을 사용해야 하는지, 몇 개의 은닉뉴런(hidden neuron)을 사용해야 하는지, 그리고 어떤 활성화함수(activation function)를 사용해야 하는지 등 신경회로망의 구조를 결정해야 할 필요성이 생긴다. Universal approximation theorem[4]은 여기에 어느 정도의 해답을 제시해주고 있다. Universal approximation theorem은 국소적으로 한계가 있고(locally bounded) 구간별로 연속인(piecewise continuous) 비다항식의 활성화함수를 가지는 한 층의 은닉뉴런으로 어떤 연속적인 함수도 학습할 수 있다는 정리이다. 즉, 한 층의 은닉층만 사용해도 문제를 해결하기에 충분하다는 것이다.

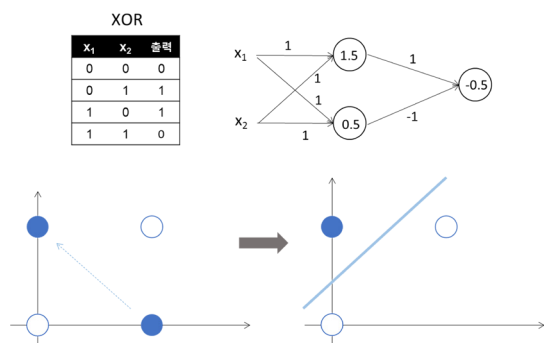


그림 4. XOR 문제 및 그 해결

신경회로망의 성능은 일반적으로 평균제곱오차(mean square error) 또는 크로스엔트로피(cross-entropy)로 측정하며 기울기 하강(gradient descent)을 이용한 오차역전파법(error back-propagation)을 통해 학습이 이루어진다. 이 때 가중치를 얼마큼 업데이트할 것인지를 결정하는 파라미터가 학습율(learning rate)로, 학습율의 값이 클수록 학습 속도는 빨라지지만 극값에 수렴하지 못하고 발산할 가능성도 높아지게 된다. 반대로 학습율의 값이 작으면 학습에 오랜 시간이 소요된다. 따라서 적절한 학습율의 값을 찾는 것도 신경회로망의 성능을 결정짓는 주요 요인 중 하나이다. 또 한 가지 학습에서 고려해야 할 것은 국부최적화(local optimization) 문제이다. 일반적인 오차 함수에는 극값이 여러 개 존재하기 때문에 기울기 하강이 시작된 위치에 따라 오차값이 전역 최적점(global optimum)에 도달하지 못하고 국부 최적점(local optimum)에 수렴하는 경우가 발생할 수 있다. 그 외에도 신경회로망을 사용할 때 고려해야 할 점들로 학습데이터를 어떤 방식으로 입력할 것인지(batch, mini-batch, online 등), 학습 속도를 향상시키기 위해서는 오차역전파법 외에 어떤 방법(conjugate gradient backpropagation algorithm[5], quasi-Newton algorithm[6], Levenberg-Marquardt algorithm[7] 등)을 사용할 수 있을지 등이 있다.

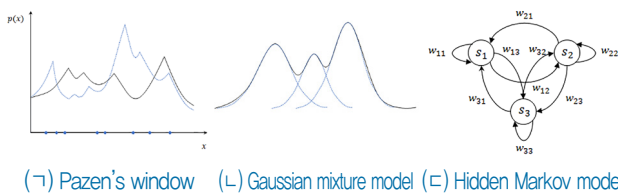


그림 5. 확률 밀도 분포 추정법의 예시

## 2. 기저벡터머신

기저벡터머신(support vector machine; SVM)은 신경회로망과 같이 결정경계를 찾기 위한 방법이다[8]. 그러나 결정경계와 각 클래스의 데이터 간의 최소 거리로 정의되는 마진을 최대화한다는 목적을 설정함으로써 일반화된 결정 경계를 찾을 수 있게 되어 신경회로망보다 다양한 문제에 대해 강인한 것으로 알려져 있다.

## 3. 확률 밀도 분포 추정법

결정경계를 찾는 신경회로망이나 기저벡터머신과 달리, 각 클래스의 확률 밀도 분포를 추정함으로써 기계학습을 수행하는 방법들도 있다. 대표적인 예로는 Parzen's window, Gaussian mixture model, hidden Markov model 등이 있다(그림 5).

데이터의 확률 밀도 분포를 추정하는 방법은 크게 파라미터를 이용한 모수(parametric) 추정 방법과 그렇지 않은 비모수(non-parametric) 추정 방법으로 나눌 수 있다. 모수 추정 방법에는 최대우도추정(maximum likelihood estimation), 최대사후확률추정(maximum a posterior estimation), EM 알고리즘(expectation-maximization algorithm) 등이 있으며, 비모수 추정 방법에는 k-NN (k-nearest neighbor), 의사결정트리(decision tree) 등이 있다.

## 4. 기계학습 모델 구축

지금까지 소개한 것과 같이, 기계학습에는 다양한 방법이 존재한다. 그러면 풀고자 하는 문제에 어떤 기계학습 방법을 적용시키는 것이 좋을까? 기계학습 방법들 중에서 어떤 것이 더 좋고 나쁨을 말하기는 어렵지만 Occam's razor에 따르면 여러 선택지가 있을 때 같은 결과를 얻는다면 그 중 가장 단순한 방법이 가장 최적의 방법이다[9].

또 한 가지 기계학습 모델을 구축할 때 고려해야 할 것은 편향-분산 트레이드 오프(bias-variance trade-off) 문제이다[10]. 일반적으로 기계학습 모델의 성능을 평가하기 위해 사용되는 평균제곱오차는 편향(bias)과 분산(variance)으로 나눌 수 있다. 편향은 모델을 학습하는 데 있어 학습 데이터를 얼마나 유연하게 받아들일 것인가에 대한 지표로 학습데이터를 충분히 유연하게 고려하지 못하면(높은 편향 값을 가지면) 기계학습 모델이 제대로 학습되지 않는 과소학습(underfitting) 문제가 발생한다. 따라서 편향 값은 데이터에 따라 모델의 정확도가 어떻게 변하는지를 측정하는 기준이 되기도 한다. 분산은 학습데이터에 대한 모델의 민감도를 나타내는 지표로, 높은 분산 값은 학습데이터에 포함된 노이즈까지 기계학습 모델이 학습했다는 의미가 되므로 과도학습 문제로 이어지게 된다. 편향과 분산은 서로 반비례적 관계를 가지고 있으므로 기계학습 모델을 구축할 때 편향과 분산 사이의 트레이드오프가 필연적으로 발생한다.

## III. 특징 추출

기계학습 과정에서 가장 중요한 것 중 하나는 주어진 문제를 풀기에 적합한 특징을 추출하는 것이다. 물론 주어진 학습데이터를 그대로 사용하는 것도 가능하지만, 여기에는 문제를 푸는데 필요한 정보 이외에도 관련 없는 다른 정보나 노이즈 등이 포함되어 있기 때문에 데이터를 가공 없이 사용하는 것은 오히려 모델의 성능을 저하시킬 수 있다. 또한 일반적으로 주어진 데이터는 높은 차원을 가지는 경우가 많기 때문에 보다 낮은 차원의 특징으로 변환할 필요성도 있다.

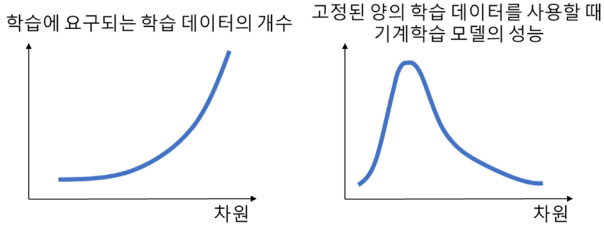


그림 6. 데이터의 차원에 따른 학습 데이터 요구량 및 기계학습 모델의 성능

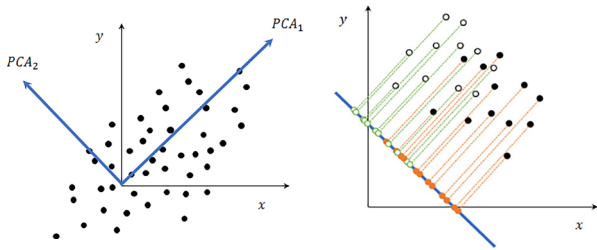


그림 7. PCA (좌)와 LDA (우)

데이터의 차원에 대해 조금 더 자세히 살펴보자. 데이터의 차원이 증가할수록 상대적으로 주어진 데이터의 희소성도 증가하게 된다. 즉, 출력 값과 데이터 간의 관계를 학습하기 위한 데이터가 부족한 현상이 나타난다. 또한 학습 대상이 되는 파라미터의 개수도 증가하기 때문에 총체적으로 학습에 더 많은 양의 데이터가 요구되는 결과를 야기한다[11]〈그림 6〉. 따라서 충분한 양의 학습 데이터가 없다면 다소 정확도가 떨어질 위험을 감수하더라도 특징을 추출하는 등의 과정을 통해 데이터의 차원을 줄이는 것이 과소학습 문제를 피할 수 있다.

특징 추출 방법에 대해 살펴보면, 먼저 선형적으로 특징을 추출하는 방법이 있다. 그 중에서도 데이터를 기반으로 특징을 추출하는 방법에는 대표적으로 PCA (principal component analysis)[12]와 LDA (linear discriminant analysis)[13]가 있다〈그림 7〉. PCA는 주어진 데이터의 정보 손실을 최소화하는 방향으로 특징을 추출하며 LDA는 주어진 데이터의 클래스 간의 거리를 최대화하는 방향으로 특징을 추출한다. 반대로 주어진 데이터를 고려하지 않고 특징을 추출하는 방법은 데이터를 다른 차원 등으로 변환한 뒤 변환계수(transform coefficient)를 취하는 형태가 일반적이다. 대표적인 예로는 discrete Fourier transform, discrete cosine transform, discrete wavelet transform이 있다. 또는 특정 도메인에 대한 사전 지식을 기반으로 특징을 추출하는 방법도 있다. 음성신호의 MFCC (Mel-frequency cepstral coefficient)[14]나 이미지의 HoG (histogram of oriented gradient)[15] 등이 해당된다.

## IV. 딥러닝

딥러닝은 패턴 인식 문제 또는 특징점 학습을 위해 많은 수의 신경층을 가지도록 모델을 구성하는 기계학습 기술들을 의미한다. 이는 실제 인간의 뇌가 뉴런들 간의 연결이 매우 깊은(deep) 구조를 가지고 있다는 점에서 보다 진보된 인공지능 기술이라 할 수 있다. 기존의 기계학습에서 다수의 신경층을 이용하는 접근은 다음의 세 가지의 문제로 인해 실질적으로는 널리 쓰이지 않았다. 첫째로, universal approximation theorem에 의거하여 하나의 은닉 신경층 만으로도 좋은 성능을 얻어내고 있었다. 둘째로, 신경층의 수를 선불리 늘렸을 때에 늘어나는 가중치의 수를 감당할 수 없었다. 셋째로, 가중치의 수가 많아졌을 때에 발생하는 과도학습 및 소위 curse of dimensionality의 위험성이 있었다. 이러한 문제들 때문에 신경층의 수를 늘릴 필요성이 대두되지 않았으나, 계산 성능, 데이터량의 증가와 알고리즘의 발달로 이러한 접근의 효용성이 밝혀지게 되었다.

딥러닝 기술이 적용되는 신경망 구조에는 자가인코더(auto-encoder)[16], 제한볼츠만기계(restricted Boltzmann machine; RBM)[17], 컨볼루션신경회로망(convolutional neural network; CNN)[18], 회귀신경회로망(recurrent neural network; RNN)[19] 등이 존재한다. 자가인코더와 제한볼츠만기계는 딥러닝 기술이 제안되면서 초창기에 제시된 모델들로, 자율학습에 용이하다는 장점이 있으나 특성상 제한적인 성능을 보인다. 현재 쓰이는 딥러닝 시스템은 대부분이 컨볼루션신경회로망이나 회귀신경회로망에 그 기반을 두고 있으며, 이들은 지도학습이 필요하지만 더욱 강력한 성능과 넓은 활용 범위로 인해 선호되고 있다. 컨볼루션신경회로망은 화상 신호 처리나 컴퓨터 비전 분야[20][21]에서 필수적인 기술이 되어가고 있으며, 회귀신경회로망은 음성 신호 처리 및 음성 인식에서 좋은 성능을 보이고 있다.

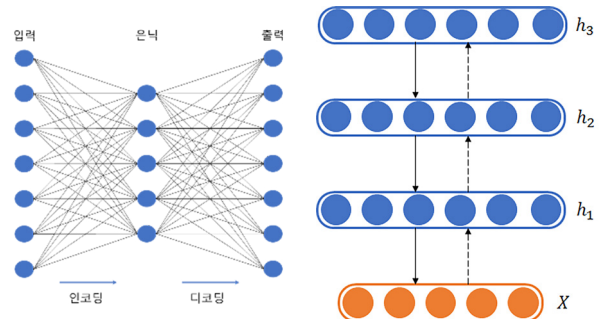


그림 8. 자가인코더(좌)와 제한볼츠만기계(우)



자가인코더[16](그림 8)는 주어진 입력을 가장 잘 복원할 수 있게 하는 방향으로 학습된다. 이러한 학습 메커니즘은 은닉 뉴런이 선형 함수에 기반할 때에는 PCA와 같다고 볼 수 있으나, 보통의 기계학습 문제에서는 비선형 함수를 이용하기 때문에 더욱 다양한 특징점을 추출하는 것이 가능하다. 자가인코더의 특징은 입력 데이터의 일부 정보를 임의로 삭제한 뒤 복원하는 방향으로 학습시키는 것으로 더욱 빠른 학습과 좋은 성능을 얻을 수 있게 되는데, 이를 잡음제거 자가인코더(denoising auto-encoder)라고 한다.

제한볼츠만기계[17](그림 8)는 확률적으로 0 또는 1의 값을 가지는 은닉 유닛을 둔 이후, 은닉 유닛이 내는 출력 값의 확률 분포를 기반으로 학습된다. 이 학습은 입력 유닛과 은닉 유닛의 결합확률 분포의 값을 최대화 시키는 방향으로 이루어지는데, 이 과정에서 대조 확산법(contrast divergence)[17]을 이용하여 계산량을 줄이게 된다.

자가인코더와 제한볼츠만기계는 입력 데이터만을 기반으로 은닉 유닛을 학습하는 자율학습 모델인데, 이러한 신경층을 여러 겹 쌓아서 딥러닝 모델을 만든 이후에는 실제로 풀려고 하는 문제와 데이터에 맞추어서 지도학습을 하는 미세조정(fine-tuning) 과정을 거친다. 이러한 형태의 모델에서 각 신경층을 자가인코더로 구성한 것은 적층 자가인코더(stacked auto-encoder), 제한볼츠만기계로 구성한 것은 심층신뢰신경망(deep belief network; DBN)이라고 한다.

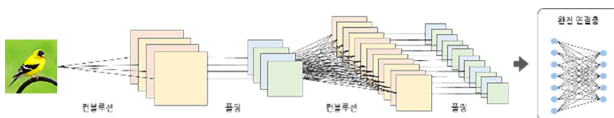


그림 9. 컨볼루션신경회로망(CNN)

CNN은 사람의 시각뉴런의 수용장(receptive field) 개념을 이용해서 만들어진 신경망[18]이다(그림 9). CNN은 컨볼루션 신경층(convolution layer), 풀링 신경층(pooling layer)과 정류선형유닛(rectified linear unit; ReLU)을 핵심 요소로 가진다. 컨볼루션 신경층은 입력과 출력을 어떤 신호의 형태로 보고 가중치를 작은 크기의 필터의 형태로 나타낸 가중치 커널이 특징이다. 즉, 아무리 입력 데이터의 차원이 높아도 커널의 크기를 작게 설정 하면 매우 적은 수의 가중치로 신경층을 정의할 수가 있다. 풀링 신경층은 몇 개의 출력 값들을 요약하셔서 데이터의 크기를 줄이는 역할을 하며, 이 과정에서 입력 데이터의 잡음이나 왜곡을 해소하는 효과를 얻을 수 있다. 정류선형유닛은 램프 함수의 활성을 가지는 비선형 뉴런으로, 기존에 신경회

로망 등에서 쓰이던 시그모이드(sigmoid) 함수가 가지고 있던 계산 부담 및 역전파 알고리즘에서의 기울기 사라짐 현상을 동시에 해결하는 효과를 가진다.

CNN의 개념은 20세기 말에 제안이 되었으나, 이것이 딥러닝 패러다임과 접목되고 딥러닝 및 기계학습의 대세가 된 것은 2012년의 ImageNet challenge 대회부터라 할 수 있다. 2011년까지는 기존의 기계학습의 방식을 이용하셔서 많은 수의 영상의 분류 작업을 실행하였는데, 2012년에 나온 CNN 기반의 AlexNet[22]이 매우 큰 성과를 거두게 되면서 그 이후의 대회에서는 CNN이 주류가 되었고, 이를 시작으로 많은 영상 관련 문제에서 CNN이 널리 이용된다.

2012년 이후로 제안되는 CNN들은 갈수록 많은 신경층을 가지게 되고, 이 과정에서 효율적으로 많은 신경층을 하나의 CNN 안에 넣기 위한 기술들이 적용된다. 2013년에는 필터 커널을 가중치의 모음으로 구성하지 않고 신경망으로 구성하는 Network-in-network(NIN)[23] 개념이 제안되었으며, 2014년에는 큰 컨볼루션 신경층을 적게 쌓는 것보다 작은 컨볼루션 신경층을 많이 쌓는 것이 계산량 감소 및 비선형 함수 근사화에 유리하다는 개념에서 착안한 VGG[24]와 GoogLeNet[25]이 제안되었다. 2015년에는 출력 값과 입력 값의 차이를 학습하여 계산량을 줄이고 신경망을 더욱 깊게 구성한 ResNet[26]이 제안되었으며, ResNet은 ImageNet challenge 대회에서 3.5%의 영상 분류 오류율을 기록하여 사람의 인지 능력인 5%보다 높은 성능을 기록하게 되었다. 최근에는 GoogLeNet과 ResNet을 결합한 형태의 신경망이 공개되었으며[32], CNN의 구조는 앞으로도 계속 깊어지고 복잡해질 것이다.

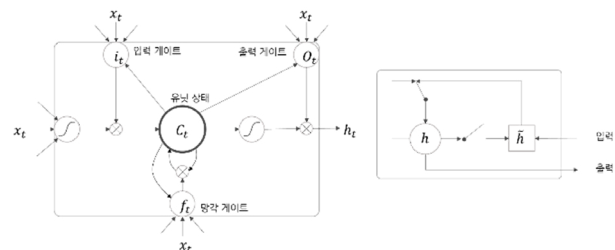


그림 10. 장단기기억(LSTM, 좌)과 게이트회귀유닛(GRU, 우)

RNN은 신경망의 출력 값의 일부가 신경망의 입력 값에 포함되는 형태의 신경회로망을 의미한다. 이러한 형태의 신경망은 시계열 데이터를 처리하는 데에 쓰이는 목적으로 예전부터 제안이 되어 왔으나, 그 특성상 오랜 시간에 걸쳐 나타나는 패턴을 잘 인식하지 못한다는 문제가 있었다. 이 문제를 해결한 것이 장단기기억(Long short-term memory; LSTM)[27]과 게

이트회귀유닛[28](Gated recurrent unit; GRU)의 도입이다(그림 10). LSTM은 가중치 이외에 기억에 관한 추가적인 정보를 셀 상태(cell state)로 저장하여 시계열 패턴의 길이를 조절한다. GRU는 LSTM에서 출력 게이트를 제거하여 간소화시킨 형태로, 더욱 단순하면서도 성능면에서는 동등하다. 이를 이용해서 제작된 RNN은 음성 신호 처리[29]나, 문자열 처리[30]에 주로 이용되고 있다.

지금까지 살펴본 딥러닝 모델들이 기계학습에 있어서 가지는 의의는 많은 수의 데이터와 많은 계산을 통해서 성능을 끌어올린 것뿐만이 아니다. 위에서 소개한 딥러닝 모델들은 어떤 문제를 푸는 과정에 있어서 사람의 개입이 필요한 특징점 추출 과정을 거치지 않기 때문에, 결과적으로 데이터 또는 문제에 알맞은 특징점을 신경층별로 스스로 추출하는 효과를 나타내게 되고, 이는 학습에 썼던 데이터셋의 범위에 따라 범용적인 특징점으로 작용할 수도 있게 된다. 이를 이용하면 이미 학습된 신경망의 일부를 이용하여 다른 문제를 푸는 데에 차용하여 학습에 요구되는 계산량을 줄일 수가 있는데, 이러한 개념을 전이 학습(transfer learning)[31]이라고 한다. 전이학습을 이용하면 딥러닝 모델을 사용하는 데에 장벽이 될 수 있는 큰 계산 요구량을 일정 부분 해소할 수 있다.

지금까지 소개한 딥러닝 모델들의 대부분은 현재의 딥러닝 패러다임이 제시되기 이전에 개발된 모델들이다. 그러나 이 모델들이 당면한 문제들을 처리하기에 적합하고 현재의 컴퓨팅 환경에 알맞아 최근 들어 그 가치가 재발견된 경우가 많다. 또한, 딥러닝 모델은 구조가 고정된 것이 아니기 때문에 그 구성을 유동적으로 설정하는 것이 가능하다. 따라서 새로운 문제가 제시되면 이에 맞추어 신경망 모델을 새롭게 개발할 수 있다. 딥마인드에서 개발한 알파고[33]와 웨이브넷[34], 구글에서 개발한 쇼&텔[35]이 그 예시로, 이들은 기존에 알려져 있던 기계학습 모델들을 결합하거나 수정해서 문제에 알맞은 새로운 모델을 제시하였다.

## V. 결론

본 논문에서는 기계학습에 대한 전반적인 내용을 설명하고 특히 최근 많은 주목을 받고 있는 딥러닝에 대해 소개했다. 기존의 기계학습은 여러 가지 모형들인 신경회로망, 기저벡터머신, PCA 등의 여러 가지 기법들이 다양하게 연구되고 이용되었다. 딥러닝 개념이 도입된 이후에는 CNN과 RNN이 패턴 인식을 위해 주로 쓰이고 있으며, 그 구조상 나타나는 특징학습을 기반으로 한 패러다임 변화가 발생하고 있다.

딥러닝의 강력한 성능과 잠재력에 힘입어, 현재 많은 연구 단체 및 IT 기업들이 딥러닝에 많은 투자를 하고 있다. 구글, 페이스북, 마이크로소프트 등 유명 기업들이 Tensorflow, Torch, CNTK 등 자사에서 개발 및 지원하는 라이브러리 및 이를 기반으로 개발한 서비스들을 제공하고 있으며, 많은 어플리케이션들이 딥러닝에 기반한 방식으로 대체되고 있다. 이러한 흐름에 따라 기계학습 분야는 매우 빠른 속도로 발전하고 있고, 따라서 기계학습이 현재 및 근미래의 IT 산업의 핵심을 쥐고 있는 분야라고 해도 과언이 아닐 것이다.

## Acknowledgement

본 연구는 미래창조과학부 및 정보통신기술진흥센터의 ICT명품인재양성사업의 연구결과로 수행되었음.(IITP-R0346-16-1008)

## 참고 문헌

- [1] McCulloch, W. A., and Pitts, W. "A logical calculus of the ideas immanent in nervous activity," Bulletin of Mathematical Biophysics, vol. 5, no. 4, pp. 115-133, 1943.
- [2] Rosenblatt, F. "The perceptron: a probabilistic model for information storage and organization in the brain," Psychological Review, vol. 65, no. 6, pp. 386-408, 1958.
- [3] Minsky, M., and Papert, S., Perceptrons, MIT Press, 1968.
- [4] Leshno, M., Ya, V., Pinkus, A., and Schocken, S. "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," Neural Networks, vol. 6, no. 6, pp. 861-867, 1993.
- [5] Charalambous, C. "Conjugate gradient algorithm for efficient training of artificial neural networks," in IEE Proceedings of Part G (Circuits, Devices and Systems), vol. 139, no. 3, pp. 301-310, 1992.
- [6] Dennis, Jr, J. E., and More, J. J. "Quasi-Newton methods, motivation and theory," SIAM Review, vol.

- 19, no. 1, pp. 46–89, 1977.
- [7] More, J. J. “The Levenberg–Marquardt algorithm: implementation and theory,” *Numerical Analysis*, Springer, 1978, pp. 105–116.
  - [8] Cortes, C., and Vapnik, V. “Support–vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
  - [9] Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. “Occam’s razor,” *Readings in Machine Learning*, 1990, pp. 201–204.
  - [10] Domingos, P. “A few useful things to know about machine learning,” *Communications of the ACM*, vol. 55, no. 10, pp. 78–87, 2012.
  - [11] Hughes, G. “On the mean accuracy of statistical pattern recognizers,” *IEEE Transactions on Information Theory*, vol. 14, no. 1, pp. 55–63, 1968.
  - [12] Jolliffe, I., *Principal Component Analysis*, John Wiley & Sons, 2002.
  - [13] Scholkopf, B., and Mullert, K. R. “Fisher discriminant analysis with kernels,” *Neural Networks for Signal Processing IX*, vol. 1, no. 1, pp. 41–48, 1999.
  - [14] Imai, S. “Cepstral analysis synthesis on the mel frequency scale,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 8, 1983, pp. 93–96.
  - [15] Dalal, N., and Triggs, B. “Histograms of oriented gradients for human detection,” in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.
  - [16] Vincent, P., Larochelle, H., Lajoie, I., Bengio Y., and Manzagol, P. A. “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion,” *Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.
  - [17] Hinton, G. E., Osindero, S., and Teh, Y. W. “A fast learning algorithm for deep belief nets,” *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
  - [18] Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. “Gradient–based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
  - [19] Mikolov, T., Kombrink, S., Burget, L., Cernocky, J., and Khudanpur, S. “Extensions of recurrent neural network language model,” in *Proceedings of 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 5528–5531.
  - [20] Dong, C., Loy, C., He, K., and Tang, X. “Image super–resolution using deep convolutional networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2016.
  - [21] Long, J., Shelhamer, E., and Darrell, T. “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.
  - [22] Krizhevsky, A., Sutskever, I., and Hinton, G. E. “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.
  - [23] Lin, M., Chen, Q., and Yan, S. “Network in network,” *arXiv preprint*, arXiv: 1312.4400, 2013.
  - [24] Simonyan, K., and Zisserman, A. “Very deep convolutional networks for large–scale image recognition,” *arXiv preprint*, arXiv: 1409.1556, 2014.
  - [25] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., and Rebinovich, A. “Going deeper with convolutions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.
  - [26] He, K., Zhang, X., Ren, S., and Sun, J. “Deep residual learning for image recognition,” *arXiv preprint*, arXiv: 1512.03385, 2015.
  - [27] Gers, F. A., Schmidhuber, J., and Cummins, F. “Learning to forget: Continual prediction with LSTM,” *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.
  - [28] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. “Gated feedback recurrent neural networks,” in *Proceedings of the 32nd International Conference on Machine Learning*, 2015, pp. 2067–2075.
  - [29] Graves, A., Jaitly, N., and Mohamed, A. R. “Hybrid

speech recognition with deep bidirectional LSTM,” in Automatic Speech Recognition and Understanding, 2013, pp. 273–278.

- [30] Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., Zhong, V., Paulus, R., and Socher, R. “Ask me anything: Dynamic memory networks for natural language processing,” arXiv preprint, arXiv: 1506.07285.
- [31] Pan, S. J., and Yang, Q. “A survey on transfer learning,” IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345–1359, 2010.
- [32] Alemi, A. (2016, Aug 31). “Improving inception and image classification in TensorFlow,” Google Research Blog (<https://research.googleblog.com/2016/08/improving-inception-and-image.html>)
- [33] Silver, D. et al. “Mastering the game of Go with deep neural network and tree search,” Nature, vol. 529, no. 7287, pp. 484–489.
- [34] Oord, A. et al. “Wavenet: A generative model for raw audio,” arXiv preprint, arXiv: 1609.03499.
- [35] Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. “Show and tell: A neural image caption generator,” In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3156–3164.

## 약 력



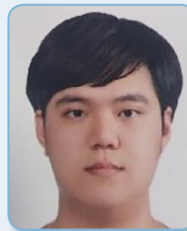
문 성 은

2013년 Chiba University 학사  
2013년~현재 연세대학교 석박통합과정 재학  
관심분야: 생체신호를 이용한 사용자 인지 분석, 기계학습



장 수 범

2014년 연세대학교 학사  
2014년~현재 연세대학교 석박통합과정 재학  
관심분야: 기계학습을 통한 멀티미디어 분석, 멀티미디어 소셜 네트워크 분석



이 정 혁

2015년 연세대학교 학사  
2015년~현재 연세대학교 석박통합과정 재학  
관심분야: 멀티미디어 신호 처리, 기계학습



이 종 석

2006년 KAIST 공학박사  
2008~2011년 EPFL 연구원  
2011년~현재 연세대학교 글로벌융합공학부  
조교수, 부교수  
관심분야: 신호처리, 머신러닝