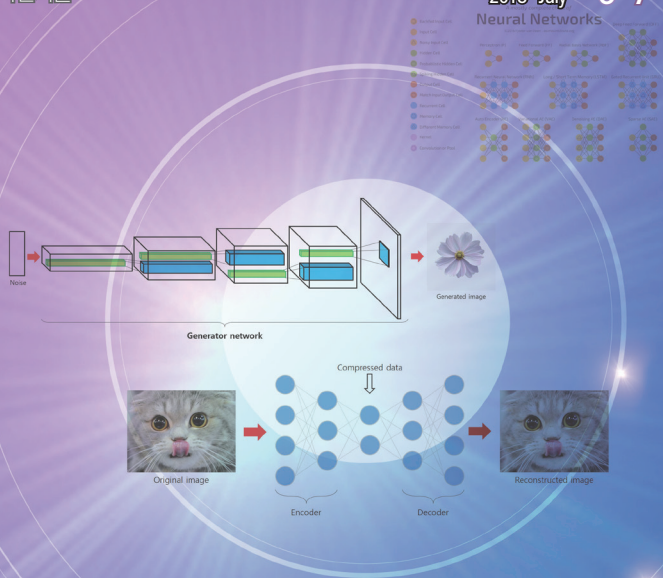
인공지능 및 딥러닝 동향



구형일 | 아주대 전자공학과 교수

1. 서 론

"지능"은 우리가 일상생활에서 흔히 사용하는 단어이지만 엄밀하게 정의하기가 쉽지 않다. 지능에 대한 정의는 연구 분야에 따라 달라질 뿐 아니라 각각의 정의는 지능의 중요한 측면을 반영하고 있다. 마찬가지로 인공지능에 관해서도 모든 사람이 동의하는 정의를 찾기 쉽지 않다. 대표적인 인공지능 텍스트북인 S. Russell and P. Norvig의 "Artificial Intelligence: A Modern Approach"에서는 인공지능의 이러한 점을 고려하여 4개의 카테고리로 인공지능의 정의를 소개하고 있다 [1]. 구체적으로 "사람처럼 vs 합리적으로", "생각 vs 행동"을 기준으로 4개의 가능한 카테고리로 인공지능을 정의하고 각 카테고리의 관점을 소개한다[1]. 물론, 이 네 가지 모두 중요한 관점이지만 이 중에서 "사람처럼 행동하는(Acting humanly)"과 관련된 기준이 공학적으로 판단하기 쉽고 공학과 관련된 연구 활동과 밀접한 관계가 있다. 이러한 관점에서의 정의는 아래와 같다.

○ 인공지능은 사람에 의해서 수행될 때 지능이 필요한 일을 수행하는 기계를 만드는 기술이다. (The art of creating machines that perform functions that require intelligence when performed by people.)

○ 인공지능은 컴퓨터가 (현재는) 사람이 잘하는 일을 할 수 있도록 하는 방법을 연구하는 학문이다. (The study of how to make computers do things, at which, at the moment, people are better.)

튜링 테스트는 이런 동작(operation) 관점에서의 정의와 관련이 있다. 튜링 테스트는 시험감독관(human interrogator)이 사람과 컴퓨터와 서면질의응답을 수행하였을 때, 응답자가 사람인지 컴퓨터 프로그램인지 구별할 수 없다면 컴퓨터가 지능을 가졌다고 봐야 한다는 요지의 테스트이다. 동작관점으로 인공지능을 정의하였을 때, 인공 지능의 구현 방식은 매우 다양할 수 있다(사실 특별한 제한이 없다). 예를 들어 전문가 시스템(expert systems)의 하나의 예인 MYCIN은 대략 450개의 규칙으로 혈액관련 진단(MYCIN to diagnose blood infection)을 수행하며 그 성능은 전문가와 비슷한 수준으로 경험이 적은 의사들보다 높은 성능을 보였다. 따라서 MYCIN은 하나의 성공적인 인공지능 시스템으로 볼 수 있다.

하지만 MYCIN의 설계과정은 이론적으로 분석하기 어려우며 다른 종류 시스템 개발에 적용하기도 어렵다. MYCIN의

1) 4 개의 카테고리: "사람처럼 생각하는(thinking humanly)", "이성적으로 생각하는(Thinking rationally)", "사람처럼 행동하는(Acting humanly)", "이성적으로 행동하는(Acting rationally)"

사용된 규칙들은 전문가와의 인터뷰를 통해 얻어졌는데 전문가들은 교과서, 다른 전문가 혹은 자신의 경험으로부터 그 규칙들을 얻었다. 또한 이러한 접근법으로는 불확실성을 다루기 쉽지 않다. 결국 전문가 시스템은 인공지능을 구현하는 하나의 방법론이지만 복잡도가 높은 문제(예, 영상판독, 번역, 음성인식 등)를 해결하기에는 한계가 있었다. 최근에는 인공지능 문제를 해결하기 위한 다양한 접근법 중에 기계학습에 기반을 둔 접근법이 큰 관심을 받고 있다.

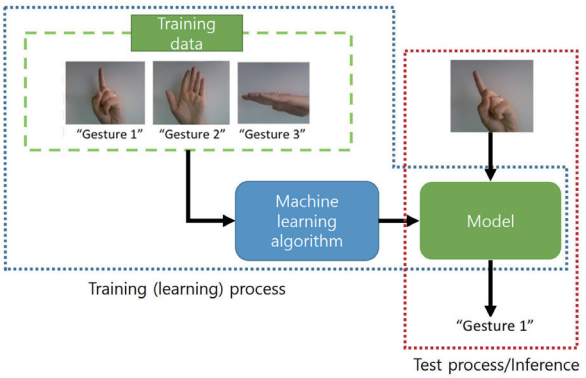


그림 1. 지도학습 과정

2. 기계학습의 개념

현재 대략 40억 개의 웹페이지가 있으며 유튜브에는 매 분당 수십 시간에 해당하는 비디오가 업로드 되고 있다 [2]. 스마트팩토리와 IoT의 결과로 각 기계들의 상태에 대한 데이터가 축적되고 있으며 수많은 의료 관련 정보들도 디지털화 되고 있다. 이와 같은 데이터의 폭발적인 증가는 데이터 분석을 일정부분 자동으로 수행하는 방법인 기계학습에 대한 수요를 증가시키고 있다. 구체적으로 기계학습은"자동으로 데이터에서 패턴을 발견하고, 이 패턴들을 새로 들어온 데이터의 라벨을 예측하거나 다른 종류의 결정을 하는 데 사용하는 방법들의 모임"으로 정의할 수 있다.

○ 기계학습 방법의 종류

- 지도학습(supervised learning, predictive learning)은 입력과 그 입력이 들어갔을 때 나와야 하는 출력의 예시를 통해 입출력 관계를 학습하는 방법이다. 아래 그림처럼 학습데이터(training data)에 기계학습 알고리즘을 적용하면 모델이라고 부르는 학습 결과물을 얻을 수 있다 (아래그림에서 파란색 점선의 내부가 학습과정이다). 학습데이터는 입력과 출력의 짝으로 이루어져 있는데 아래의 예에서는 손을 촬영한 영상이 입력, 손의 자세에 대한 라벨이 출력이다. 학습이 완료되어 얻게 된 모델을 사용하여 (트레이닝 셋에 포함되지 않은) 새로운 입력에 대해서 라벨을 예측할 수 있다 (아래 그림에서 붉은 색 점선 내부). 지도학습은 기계학습의 대표적인 예로서 우리가 아는 많은 시스템이 이와 같은 방식으로 학습되었다.
- 비지도학습(unsupervised learning)은 라벨 없이 데이터만 주어지는 경우, 데이터로부터 데이터 내부의 숨겨진 구조를 발견하는 방법이다. 구체적으로"군집화(clustering)","차원축소(dimension reduction, feature learning)", "생성모델학습(generative model learning)"의 일을 수행할 수 있다. 군집화는 데이터를 유사도에 따라서 여러 개의 그룹으로 나누는 작업이다. 이 과정에서 데이터 상의 거리를 어떻게 정의하느냐에 따라서 다양한 군집 결과가 나올 수 있으므로 문제의 목적에 맞는 거리를 정의하는 것이 매우 중요하다. 물론 이러한 거리를 정의하는 데는 도메인 지식이 필요하다.

차원축소(특징학습) 역시 비지도학습의 예로서 라벨 없이 수행할 수 있다. 일반적으로 오토인코더는 입력과 출력이 같도록 내부 파라미터를 학습한다. 단, 모델 내부에 병목(bottleneck) 구조를 만들어 높은 차원의 입력이 낮은 차원의 벡터로 표현되었다가 다시 복원 (reconstruction) 되도록

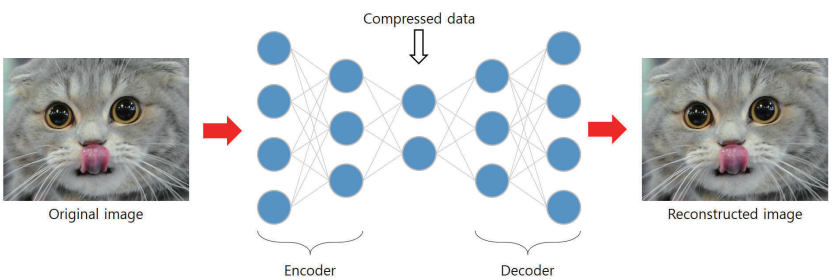


그림 2. 차원축소 예시

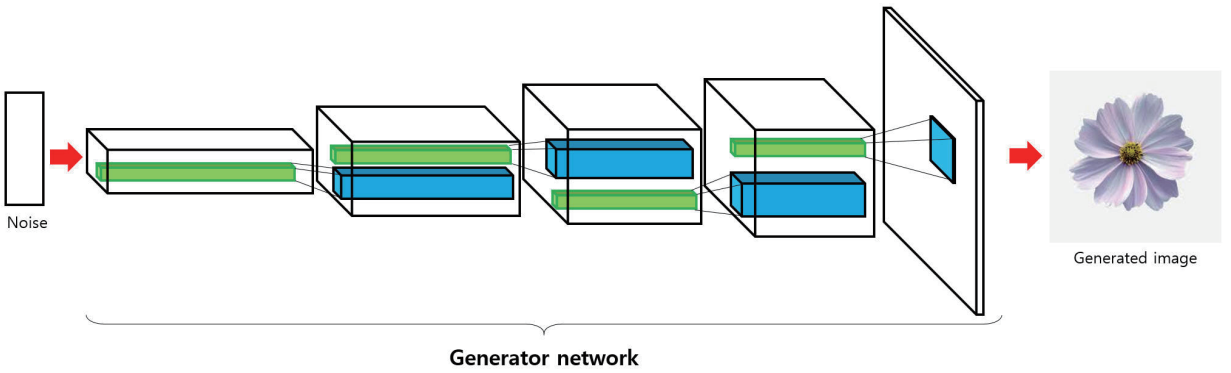


그림 3. 생성 모델 예시

한다. 이와 같은 방법으로 차원축소를 수행하면 데이터의 중요한 구조를 발견할 수 있으며 이후 데이터 처리에 도움을 준다. 라벨이 주어진 데이터가 많지 않은 상황에서 이와 같은 차원 축소 알고리즘을 통해 데이터 벡터의 차원을 줄이는 방식으로 많이 사용되곤 한다.

생성모델학습은 데이터로부터 그와 유사한 데이터를 추가적으로 생성할 수 있는 확률 모델을 학습하는 방법

생성모델학습은 데이터로부터 그와 유사한 데이터를 추가적으로 생성할 수 있는 확률 모델을 학습하는 방법으로 비지도학습의 또 다른 예이다. 일반적으로 특정 분포를 따르는 난수를 입력하면 학습 데이터와 유사한 데이터를 생성해준다. 예를 들어, 많은 수의 꽃 이미지로 네트워크학습을 수행하면 아래와 같이 잡음로부터 꽃 이미지를 생성하는 일을 할 수 있다. 최근에는 GAN(Generative Adversarial Network)을 이용한 방식이 많은 주목을 받고 있다. GAN 샘플을 생성하는 신경망과 합성된 이미지와 실제 샘플을 구별하는 신경망을 경쟁적(적대적으로)으로 학습시키는 방법이다.

- 강화학습(reinforcement learning)은 어떤 환경에서 정의된 에이전트가 현재의 상황을 입력(observation)으로 받아 누적 보상(reward)이 최대가 되는 액션을 선택하는 방법을 학습하는 방법이다. 강화학습은 지도학습과 비교하여 일반적으로 적용 가능한 인공지능(general purpose artificial intelligence)에 가까우며 현재 활발한 연구가 진행되고 있다. 딥마인드에서 강화학습을 이용하여 다양한 Atari 게임을 수행할 수 있는 알고리즘(deep Q-learning)을 개발하였다. 강화학습은 학습 과정에 라벨이 있는 데이터가 필요 없다는 점은 지도학습과는 거리가 있다. 대신 강화학습은 액션을 취하고 그 상황 변화와 관찰하고 리워드를 환산할 수 있는 환경(예, 시뮬레이터) 이 필요하다. 바둑 인공지능인 알파고에도 강화학습이 적용되었다. 하지만 강화학습 연구는 시뮬레이터가 적용되기 어려운 분야가 많다는 것과 적절한 보상 함수의 설계가 쉽지 않다는 어려움이 있다.

3. 인공신경망의 개념

뇌는 뉴런이라는 단위로 구성되어 있다. 이런 생물학적인 모델에서 영감을 받아 개별 뉴런을 수학적으로 모델링하고, 이러한 인공 뉴런 다수를 연결하여 인공신경망을 구성하여 기계학습을 수행하려는 접근법이 있다. 구성된 인공신경망에는 많은 수의 파라미터(시냅스의 가중치)가 포함되어 있는데 이러한 파라미터는 역전파 알고리즘(back-propagation algorithm)을 이용하여 찾는다. 구체적으로 역전파 알고리즘은 목적함수를 최적화하기 위해 파라미터를 업데이트 하는 방법이다. 이 방법은 현재 위치 (현재 파라미터) 주변의 값을 일차식으로 근사하고 함수 값이 줄어드는 방향으로 파라미터 값을 업데이트하는 작업을 반복하는 알고리즘이다. 단, 신경망의 각 모듈 들이 파라미터 업데이트와 관련된 신호를 추론 과정과 비교하여 역순으로 전파하는 방식으로 구현되기 때문에 역

전파 알고리즘이라 불린다. 이 방법은 전역 최적값(global optima)이 아니라 국소 최적값(local optima)에 갇힐 수 있다는 약점이 있지만 매우 큰 규모의 학습 데이터를 처리할 수 있으며 실험적으로 효과적임이 알려져 있다. 이와 같이 내부 파라미터를 찾는 과정을 학습 (training)이라고 부른다. 역전파 알고리즘을 통해서 학습된 신경망은 새로운 데이터를 처리하는 데 사용되며 이 과정은 추론(inference)이라고 한다. 인공신경망 기반의 접근법은 다른 기계학습 방법과 비교하여 다음과 같은 장점을 가지고 있다.

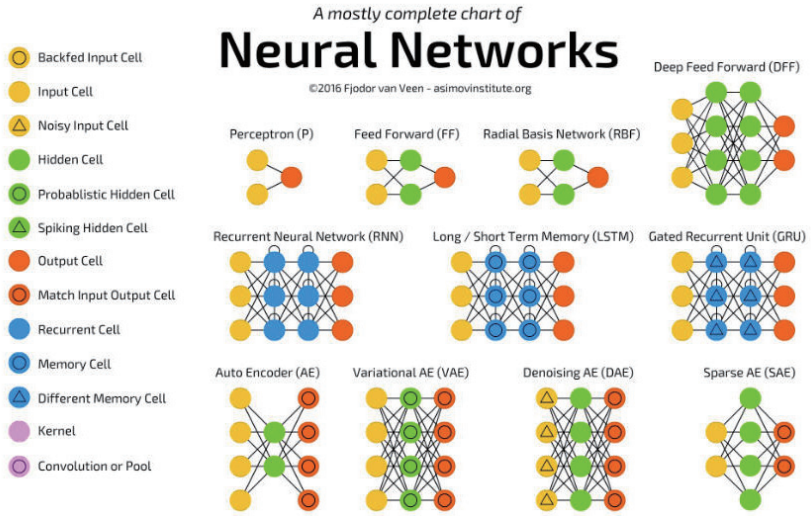


그림 4. 다양한 신경망 (일부만 소개) [3]에서 캡쳐

- 인공신경망은 매우 다양한 형태의 함수를 근사할 수 있다(universal approximation theorem). 이 정리는 조건을 만족하는 파라미터를 결정할 방법이 아니라 존재성을 이야기하는 것이지만 어떤 입출력 관계이든 인공신경망으로 표현할 수 있다는 수학적인 정리는 인공신경망이 일반적인 문제를 다룰 수 있는 표현력을 가지고 있음을 보여준다.
- 대규모 데이터에 적용할 수 있는 학습알고리즘이 존재한다. 인공신경망을 학습하기 위해서 일반적으로 사용되는 방법은 역전파 알고리즘의 하나의 구현 예시인, SGD(Stochastic gradient descent)이다. 이 방법은 작은 크기의 배치(mini-batch)에 대해서 그레디언트를 계산하고 이를 이용하여 내부 파라미터를 업데이트하는 방법이다. 경험적으로 작은 크기의 배치(batch)에 대해 적용된 SGD가 좋은 일반화 성능(generalization performance)을 보이며 대규모 학습에도 적합함이 알려져 있다.
- 다양한 업무를 처리할 수 있도록 인공신경망 구조를 설계할 수 있다. Feed forward 구조를 적용한 지도학습이 가장 많이 사용되고 있지만 차원축소, 생성모델, 강화학습 등에 사용되는 모델을 구성할 수 있다. 지도학습의 경우는 물체 검출, 영역 분할(semantic segmentation)등 영상처리에 적합한 인공신경망 구조가 제안되어 있으며 시퀀스(비디오, 문장 등)를 처리하기 위해 적합한 인공신경망 구조도 개발되어 있다. 실제로 아래 그림에서 볼 수 있듯이 매우 다양한 형태의 구조가 제안되어 있으며 인공뉴런은 이런 유연성(flexibility)을 지원한다.
- 병렬처리에 매우 적합한 구조이다. 인공신경망은 매우 단순한 구조를 가진 기본단위 (인공 뉴런)가 규칙적으로 배치되어 있는 구조이다. 따라서 하드웨어로 구현하거나 병렬처리를 하기에 매우 유리하다. 학습과정인 역전파 알고리즘의 경우도 비슷한 방식으로 표현할 수 있다.

4. 딥러닝의 개념과 특징

다층구조의 신경망(multi layer perceptron, artificial neural network)과 그러한 신경망을 학습시키는 역전파 알고리즘은 수십 년 전부터 알려져 있었다. 하지만 깊은 신경망(deep neural network, 레이어가 많은 인공신경망)의 학습은 오랜 시간동안 어려운 일로 받아들여졌다. 경험적으로 보았을 때, 깊이가 4~5를 넘어가는 신경망을 학습하면 일반화 성능이 좋지 않았기 때문이다. 하지만 최근 데이터의 증가, 병렬처리 하드웨어의 발달, 학습 알고리즘의 발전으로 매우 깊은 신경망(수십~수백)의 학습이 가능해졌으며, 많은 문제에 깊은 신경망을 적용한 결과 기존방법과 비교하여 좋은 성능을 보이고 있다. 특히 “Deeper is better”라는 표현이 있을 정도로 이런 경향성은 일관되게 보고되고 있다.

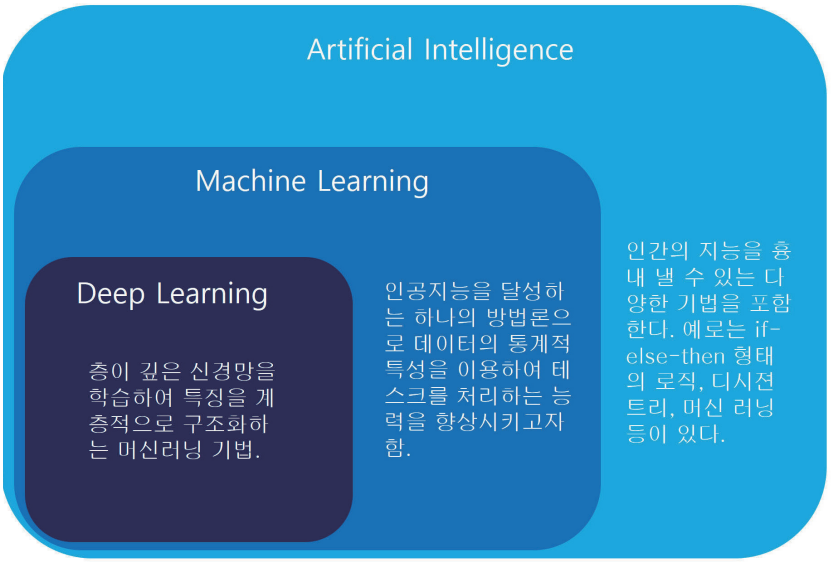


그림 5. 딥러닝, 기계학습, 인공지능 비교

인공지능, 기계학습, 딥러닝의 관계를 정리한 그림은 아래와 같다. 딥러닝은 규모가 큰 신경망을 사용했다는 점과 방대한 양의 데이터로 학습한다는 특징을 가지고 있는 기계학습 알고리즘이고 기계학습은 인공지능의 하나의 구현 방식이다. 하지만 딥러닝 알고리즘은 기존 인공지능, 기계학습 방법과 비교하여 결정적인 특징 몇 가지를 가진다.

○ 딥러닝은 데이터의 양에 비례하여 성능이 향상 되는 경향성을 보인다. 전통적인 기계학습 알고리즘은 데이터로부터 특징(feature vector)을 설계하고 이를 활용하여 비교적 단순한 기계학습을 수행하였다. 이는 적은 수의 데이터에도 불구하고 동작하는 알고리즘의 개발을 가능하게 하지만 데이터의 양이 증가하더라도 성능이 정체되는 경향을 보이게 된다. 반면에 딥러닝 기반의 접근법은 데이터가 공급됨에 따라 성능이 꾸준히 향상되는 특성을 보인다. 이는 데이터가 풍부한 현시점에 기존 방법과 비교하여 매우 중요하고 현실적인 장점이다.

○ 딥러닝 학습의 결과로 얻어진 모델을 분석하면 층이 높아질수록 점진적으로 복잡한 특징을 구성함을 확인 할 수 있다. 영상인식에서 딥러닝이 좋은 성능을 보이는 이유를 분석하기 위해서 학습된 모델을 분석하는 다양한 연구가 수행되었다. 이러한 연구 결과에 따르면 이미지를 처리하는 깊은 신경망의 밑의 레이어는 방향성 에지와 같은 단순 패턴에 반응하는 뉴런으로 구성되어 (학습되어) 있지만, 순차적으로 코너, 물체의 부분, 물체의 전체적인 모양 등의 복잡한 패턴에 반응하는 뉴런으로 구성되어 있다. 이처럼 영상의 특징을 잘 표현하는 방법을 배운다는 점을 강조하여 딥러닝의 특징을 "표현법 학습(representation learning)"이라고 부르기도 한다. 기존의 기계학습 방법이 각 도메인별로 좋은 표현법을 엔지니어(도메인 전문가)가 설계해야 했다는 점과 비교하여 딥러닝은 좋은 표현법을 데이터로부터 찾을 수 있도록 하며 도메인 지식이 적게 의존한다는 장점이 있다.

자연어 처리(natural language processing)에서도 이러한 표현법 학습은 유용하다. 구체적으로 자연어 처리 과정에서 과거에는 각 단어에 고유 인덱스를 할당하여 단어를 표현하였다. 이 방법은 수학적으로 명확하지만 단어의 의미가 표현법(representation)에 반영되지 않는다는 한계가 있다. 딥러닝을 이용하는 최근 접근법에 따르면 단어들은 수백 차원의 벡터로 표현 되는데(word embedding) 이 벡터값 들이 단어가 가지는 의미를 반영하게 된다. 예를 들어, 숫자를 표현하는 단어(one, two, three 등)들은 수백차원의 벡터공간에서 비슷한 곳에 모인다거나 특정 축이 성별(gender)을 나타내기도 한다. 딥러닝은 이러한 표현법을 데이터로부터 자동으로 학습하는 것을 가능하게 하며 이를 딥러닝 접근법이 좋은 성능을 보이는 가장 중요한 원인으로 지목된다.

○ 딥러닝의 또 다른 특징은 end-to-end 학습이 가능하다는 점이다. 전통적인 접근법은 특징 벡터의 설계를 비롯하여 시스템 설계과정에서 엔지니어가 개입하여 경험이나 직관에 의존하여 설계해야 하는 많은 요소를 가지고 있었다. 예를 들어 글자 인식 시스템(Optical Character Recognition System)의 경우 문자 이미지 입력에 대하여 이진화(binarization), 단어 단위의 분리(word segmentation), 글자 단위의 분리(character segmentation)를 순차적으로 수행한 후 개별 글자(character recognition)를 인식하는 파이프라인을 주로

사용하였으며 각 단계의 알고리즘은 독립적으로 개발 적용 되었다. 이런 접근법은 상대적으로 적은 양의 데이터로도 개발 가능하다는 장점이 있지만, 개별 단위의 개선의 합이 전체 성능의 개선과 명확한 관계를 가지지 않으며 개인의 직관적인 결정에 크게 의존할 수밖에 없다는 문제점이 있다. 또한 전체 시스템의 성능 향상을 위하여 추가로 어떤 작업을 해야 할 지 분석하기 쉽지 않다.

이와 다르게 딥러닝은 이러한 요소를 줄이고 입력과 출력의 예시만으로 전체 시스템의 학습을 가능하게 하고 있다. OCR 이외에도 음성인식이나, 번역, 이미지/비디오 캡션닝(image/video captioning) 등이 중요한 예이다. 음성 인식의 경우 음성 신호를 입력, 인식된 결과물(예, 텍스트)을 출력물로 하여 학습할 수 있으며, 번역의 경우는 A 언어로 작성된 문장과 그 문장에 대응되는 내용을 B 언어로 작성된 예시로 학습할 수 있다. 이미지와 비디오와 그 내용에 대응되는 기술(description)이 적힌 데이터로부터 전체 캡셔닝 시스템의 학습이 가능해지고 있다.

5. 딥러닝의 한계

딥러닝 기술은 현재 활발한 연구 개발이 진행되고 있는 분야로 기술 자체의 한계를 이야기하는 것은 성급할 수 있다. 하지만 (현 시점에) 딥러닝 기술을 실제 문제에 적용할 때 겪게 되는 어려움은 다음과 같이 정리될 수 있다.

○ 빅데이터가 필요하다. 깊은 신경망의 학습을 위해서는 많은 양의 데이터, 그리고 많은 경우에 전문가에 의해서 분류된 양질의 데이터가 필요하다. 이러한 데이터를 얻는 과정에는 비용이 많이 들며, 데이터를 구하는 것이 현실적으로 불가능할 때가 많다 (희귀 질환 관련 의료 데이터, 스마트 팩토리에서 고장사례 등).

○ 경험적으로 결정해야 요소가 여전히 많다. 딥러닝이 데이터로부터 패턴을 학습하는 방법이기는 하지만 아직은 개발자가 경험이나 직관을 바탕으로 결정해야 할 변수(하이퍼-파라미터)가 많으며 이 변수들을 변경해가며 학습하는 과정에는 많은 시간과 비용이 필요하다. 대표적으로 개발자가 결정해야 하는 요소로는 네트워크의 구조가 있으며, 학습과정에서 학습률(learning rate), 배치 사이즈(batch size), 모멘텀(momentum) 등을 결정해야 한다. 각 조합에 대해서 결과를 확인하기 까지 수시간에서 수주에 이르는 학습 시간이 필요하며 이는 개발 과정의 병목으로 작용한다.

○ 강화 학습의 경우 보상 함수(reward function)설계가 쉽지 않다. 바둑의 경우는 보상 함수가 승/패로 명확하지만 일반적인 상황에서 보상 함수 설계는 명확하지 않으며, 간단한 게임에서 조차도 설계자의 의도와 다르게 동작하는 경우도 보고되어 있다 [4].

참고문헌

[1] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach. NJ, USA: Prentice Hall Press, 2009.

[2] http://www.worldwidewebsize.com/

[3] http://www.asimovinstitute.org/neural-network-zoo/

[4] https://blog.openai.com/faulty-reward-functions/