코딩교육 열풍과 현주소 - 5

인공지능 기초와 체계적인 데이터 관리

글. 김승욱 Rloha 대표, 데이터 분석 교육 및 컨설팅
'빅데이터 분석, R좀 R려줘', 'R 데이터 분석' 등 관련 칼럼 및 강의 진행

이전 글에서 인공지능을 위한 데이터 수집을 다루었다. 사실 어떤 데이터를 수집할지 결정하는 것은 어떤 인공지능을 구현할지 논의가 된 다음에 결정하는 것이다. 예를 들어 환자의 상태를 진단하는 인공지능을 만든다고 하자. 대표적으로 다음과 같은 데이터를 기반으로 구현할 수 있을 것이다.

진료 기록

	A	B	C	D	E	F	G
1	region	mk_11_1	mk_11_2	mk_145_1	mk_145_2	mk_BT4_1	mk_BT4_2
2	16'IS1-1	147	147	0	0	0	0
3	16'IS1-2	169	181	173	175	293	295
4	16'IS1-3	169	181	173	175	293	300
5	16'IS1-4	169	181	173	175	293	300
6	16'IS1-5	147	147	173	175	300	300

그림 1. 정형 데이터 예시

단순하게 말하면 엑셀에 잘 정리된 데이터는 전문적인 용어로 정형 데이터(structured data)라고 한다. 이런 데이터는 이미지나 영상 같은 비정형 데이터(unstructured data)보다 상대적으로 처리가 용이하기에 정형 데이터를 기반으로 프로젝트를 한다면 다른 데이터에 비해 비교적 수월하게 진행할 수 있을 것이다.

의사 소견서

여러 항목에 표기하는 것도 있겠지만, 이 경우는 의사가 환자의 상태를 진료하고 작성한 글을 기준으로 한다.

비슷하게는 특정 방송 프로그램이나 영화의 평가 및 후기 같은 글이 되겠다. 요즘에는 전산화 시스템이 잘 되어있지만, 만약 그렇지 않은 경우는 손글씨를 인식하는 인공지능까지 만들어야 하는 난관이 기다리고 있다. 일단, 시스템에 입력된 데이터를 분석한다고 해보자. 사람이 사용하는 글자는 고유의 문법 체계를 가지고 있기에 컴퓨터가 잘 해석할 수 있도록 처리를 잘해주어야 한다. 그렇기에 사람이 작성한 글과 같은 자연어(Natural Language)를 처리하는 것을 자연어처리(NLP, Natural Language Processing)라고 부르며 추가로 전문적인 지식이 필요한 경우가 많고 오탈자, 띄어쓰기, 신조어, 전문용어 때문에 시간과 노력이 많이 들어간다. 그래서 관련 인공지능 서비스인 챗봇(chatbot)의 구현이 어렵다.

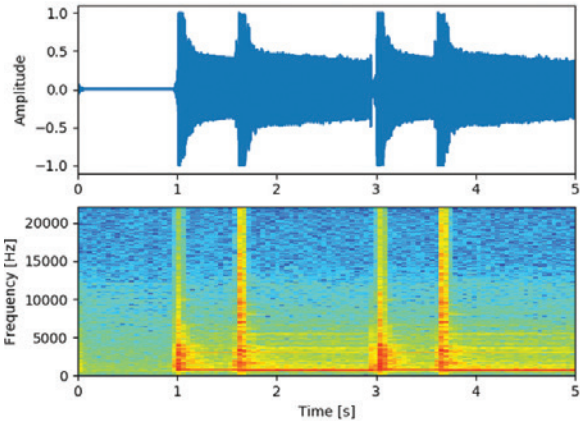


그림 2. 음성 신호(위)와 음성 분석에 활용되는 스펙트로그램(아래) 예시

그림 3. MS face API 사용 예시(서미소랑 선생님)

환자 음성

의사 소견서 같은 자연어처리 보다 한 단계 더 난이도가 높다. 한 단계라고 하긴 했지만 절대 쉽게 보면 안 된다. 음성은 신호이다. 기본적으로 주변 소음이 섞이면 사람의 음성과 분리하는 작업을 우선시해야 하며, 인공지능 학습에 사용할 음성을 추출하려면 여러 사람 중에서 특정 한 사람의 음성만 추출하는 것도 꽤 어려운 기술이다. 게다가 사람이 말을 하게 되면 개인의 특성이나 감정에 따라 목소리 톤이 꽤 다르며, 사투리를 쓰는 경우도 있다. 이렇게 변수가 많아 음성 데이터 기반 인공지능 구현에는 매우 큰 비용이 발생한다.

환자 사진

인공지능이 가장 큰 성과를 보이는 것이 이미지/영상 분야이다. 그리하여 영상에서 몇 분 몇 초에 어떤 연예인이 등장하는지 표기해보는 개인 프로젝트를 도전하는 사람도 있고 해당 프로젝트를 구현하기 위한 참고자료도 꽤 많은 편이다. 영상은 여러 장의 이미지가 모인 결과물에 가깝고, 가장 기본이 되는 것은 한 장의 이미지를 잘 분석하는 것이 되겠다. 데이터의 형태가 어떠하던 결국에는 숫자로 바꿔야 한다. 보통 이미지 기반 데이터를 활용한 인공지능을 구현하기 위해서는 음성의 잡음처럼 이미지의 불필요한 부분을 잘라내는 작업도 필요하고, 이미지 크기를 균일하게 만드는 작업도 필요하다. 그 이후에 해당 이미지의 특징을 추출하여 정리하는 등 각종 절차가 수반된다.

관련 분야의 모든 자료를 다 활용하여 거대한 시스템이나 서비스를 만들면 좋겠지만 이는 불가능에 가깝다. 왜냐하면 3~5년 이상의 중장기 프로젝트를 1년 만에 하라고 재촉하는 상황이 많기 때문에 그냥 “안된다.”라고 생각하면 편할 것이다. 물론 자본주의 사회라 그만큼 돈을 투자하면 관계없는데 과연 아낌없이 투자하는 조직은 몇이나 될까? 가용한 데이터 목록 중에서 하나 꼽으라고 한다면 프로젝트 시작 전에 진지하게 해야 한다.

자료는 충분한가?

흔히 말하는 딥러닝(신경망 기반 인공지능)은 수천 개 이상의 데이터가 있을 때 충분한 성능을 발휘한다고 한다. 적은 수의 데이터로도 충분한 성능을 내는 인공지능을 만들기 위해서 여러 연구자가 노력은 하고 있지만, 그래도 자료가 풍부하면 성능 좋은 인공지능 모델을 만들기 수월해진다.

시간은 충분한가?

기존의 통계/머신러닝 모델과 달리 딥러닝 모델[1]은 만드는 과정에 공이 많이 들어간다. 마치 돌을 깎아서 조각을 하는 것과 비슷하다고 보면 된다. 어떤 데이터를 선택하느냐는 마치 어떤 돌을 깎을까 고민하는 과정이라고 할 수 있다. 데이터만 선택하면 다 끝나는 것 같지만 그 뒤에 산더미 같은 일이 많다. 즉, 딥러닝 모델을 구현하고자 하는 데이터 종류를 다뤄본 사람이 없다면 프로젝트 기간을 좀 더 길게 잡아야 할 수 있다.

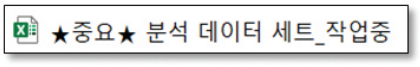
기술 구현은 가능한가?

시간 문제와 엮이는 사항이다. 기술력이 부족하다면 최종 산출물의 질적 저하와 프로젝트 수행 기간의 불확실성이 커짐은 자명하다. 각 데이터 특징에 맞는 딥러닝 모델의 종류는 생각보다 많으며, 요리사라고 해서 한식/중식/양식/일식 등 모든 음식을 잘할 수 없는 것처럼, 단순히 인공지능 전문가라고 해서 모든 종류의 딥러닝 모델에 능통한 것은 아니다. 무작정 프로젝트를 강행하는 것은 돌이킬 수 없는 결과를 야기할 수 있다.

어떤 데이터를 사용할지 충분한 고려를 하고 수집을 했다고 하자. 해당 데이터를 잘 활용하기 위해서는 체계적인 관리가 필요하다. 마우스 클릭만으로 가능한 여러 전문 상용 툴을 사용하지 않고 코드를 사용하여 연구를 수행하는 것은 연구의 재현성에 의의를 두는 경우가 많은데, 분석 데이터의 체계적인 관리가 이루어지지 않으면 다음 연구에서 해당 데이터를 제대로 활용하지 못할 가능성도 높고 재현이 어려울 수 있다. 그렇기에 당장은 다소 귀찮을 수 있지만, 멀리 보고 신경 써서 관리하는 방법을 고민해보도록 하자.

파일명

꼭 인공지능에 국한되는 것이 아니라 일반 업무에서도 체계적인 파일명 관리는 중요하다. 특히 이미지 파일의 경우 많게는 수십만 개의 파일이 있는데 제대로 명명하지 않으면 자칫 학습에 잘못된 데이터가 끼어들어 가 처음부터 다시 학습해야 할 수 있다. 물론 문서 작업만 하는 경우에는 다음과 같은 파일명이 괜찮을 수 있다.



하지만 데이터 분석용으로 사용할 파일은 이렇게 관리하면 굉장히 곤란한 상황을 맞이할 수 있으며 이 경우 다음과 같은 원인으로 문제가 발생할 수 있겠다.

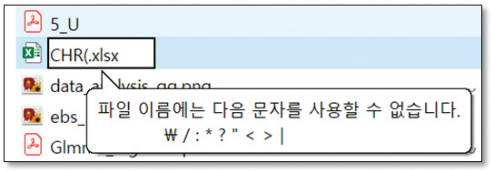


그림 4. Windows의 파일명 경고문

- 허용하지 않는 특수문자
- 한글
- 파일 사용처
- 버전관리

특정 프로그램이나 운영체제의 경우 특정 특수문자를 허용하지 않을 수 있다.

그리고 한글의 경우 운영체제별로 인코딩이 달라 글자가 깨져 보이거나 처리가 까다로울 수 있다. 그래서 분석할 데이터가 들어있는 파일은 되도록 숫자와 영어 사용을 권장하며, 띄어쓰기도 언더바(_)로 대체하는 것이 좋다. 파일 사용처를 언급한 것은 각 데이터가 어떤 분석에

그림 5. 왕초보영어 프로그램 배너 및 파일

1) 인공지능(딥러닝) 또한 기계학습(Machine Learning)에 들어가나 여기서 기계학습은 인공지능 모델(수식 등)을 제외한 나머지를 지칭한다.

사용되는지, 어떤 내용이 들어있는지 알 수 있는 단어가 있으면 좋다. 목록의 네 번째에 있는 버전 관리는 각 파일의 생성 또는 작업 날짜만 적어주어도 충분하다.

목록형

	A	B	C	D	E	F	G
1	파일명	사람	남자	여자	가로	세로	용량
2	img_ppl_0001.jpg	2	1	1	200	300	1200
3	img_ppl_0002.jpg	1	1		400	600	4800
4	img_ppl_0003.jpg	3	0	3	250	600	3000
5	img_ppl_0004.jpg	2	2	0	1280	800	20480

그림 6. 파일 정리 예시

특히 이미지나 영상 데이터를 기반으로 인공지능을 구현하는 경우는 파일 개수가 제법 많기도 하고 각 파일은 다른 특징을 담고 있다. 예를 들어서 앞의 그림에 사람이 2명 있다면, 파일명에 p2라고 적던지 특정 정보를 담을 것이다. 하지만 이런 정보가 많게 되면 파일명에 모두 담기 어려우므로 다음과 같이 정리하는 것을 권장한다.

이제 데이터 수집과 정리까지 알아봤으니 인공지능이 무엇인지, 어떻게 공부해야 하는지 알아보도록 하자. 인공지능이라고 하면 그 경계가 모호하다. 이는 마치 빅데이터는 얼마나 커야 빅데이터라고 불러야 하는지 고민하는 것과 비슷하다. 이전에는 데이터가 크다고 하면 100MB도 크다고 할 수 있지만, 필자가 느끼는 큰 데이터는 10GB 정도 된다. 이렇게 각자가 느끼는 체감 데이터 용량도 다르고 시대가 변함에 따라 다룰 수 있는 데이터의 크기도 달라 절대적인 기준을 세우기 어렵다. 그리고 인공지능이라는 것도 1700년대 즈음에는 파스칼 계산기(Pascal's calculator)를 인공지능이라고 지칭했을 수 있다.

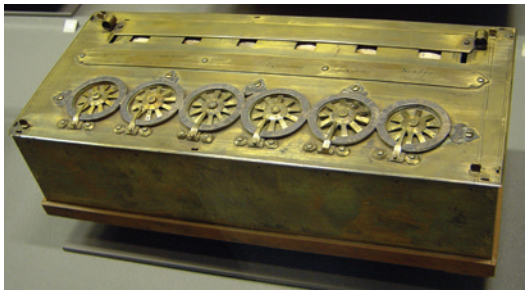


그림 7. 1652년에 제작된 파스칼 계산기

요즘은 인공지능이라고 하면 대체로 다층신경망 구조로 학습을 하는 딥러닝 기반으로 구현된 산출물을 뜻한다고 보면 된다. 가끔 억지를 부리는 경우는 알아서 혼자 잘 동작하는 프로그램이나 장치를 인공지능이라고 하는데 말 그대로 인공지능(人工知能)이겠지만 일반적으로는 이를 인공지능이라고 부르지 않는다. 그렇다면 여기서 언급된 다층신경망은 무엇인가? 다층신경망을 영어로 적어보면 Multi Layer Neural Network로, 층이 많다는 뜻의 Multi Layer를 깊다는 표현을 사용하여 Deep이라고 바꿔 부르기도 한다. 그리하여 Deep Neural Network라는 표현을 쓰고 줄여서 DNN으로 표기하기도 한다. 여기서 가장 핵심이 되는 것은 신경(Neural)이 되겠다. 이것이 그물망을 이루며 이런 그물망이 다층구조로 쌓여 있는 모양새이기 때문이다.

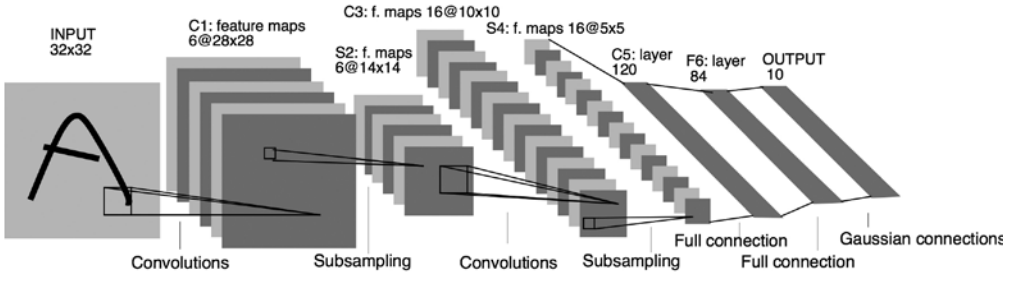


그림 8. 다층신경망 구조 예시

그렇다면 여기서 말하는 신경(Neural)은 무엇이고 왜 그물망처럼 연결이 되어야 하는가? 우선 신경부터 알아보도록 하자.

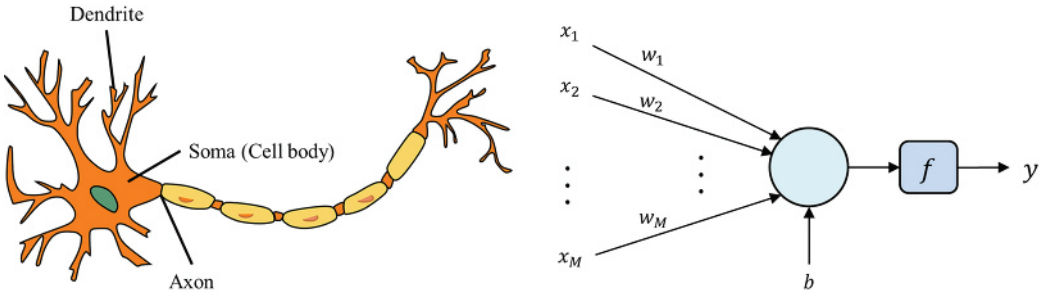


그림 9. 뉴런(좌)과 퍼셉트론(우)

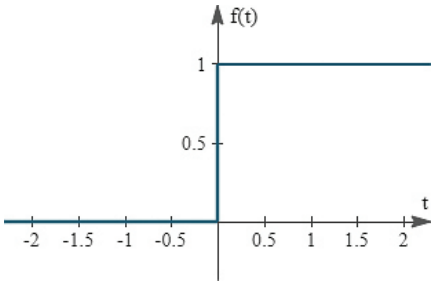


그림 10. 단위 계단 함수

신경의 경우 전기 신호가 입력되면 입력 전기 신호의 강도에 따라서 출력값이 조정된다. 예를 들어 입력이 0 이상이면 1, 0 미만이면 0으로 출력되는 경우는 간단하게 [그림 10]과 같이 그래프로 표현할 수 있다.

신경에 입력된 값이 많겠지만, 그 값을 종합한 결과가 0 또는 1 두 가지로 표현될 수 있다는 것을 알 수 있다. 같은 규칙을 가진 신경이 많다면 어떻게 될까? 단지 두 종류의 출력이 아닌 훨씬 다양한 출력을 낼 수 있지 않을까? 그래서 연구자들은 이 신경을 여러 개 연결하여 그물망 구조(Network)를 만들고 이마저도 부족하여 그물망을 여러 층 쌓아서 신경망 구조를 설계한다. 그리고 각각의 신경에 적용되는 규칙을 변경하여 더욱 다채로운 출력값이 나오도록 설계하기도 한다. 여기서 말하는 규칙이 활성함수(Activation Function)이다.

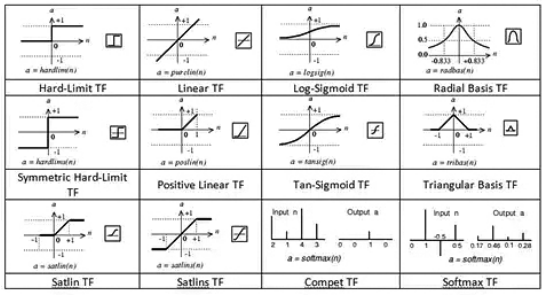


그림 11. 다양한 활성함수

인공지능 연구자는 보통 딥러닝 모델의 성능을 최대한으로 끌어올리기 위해서 여러 가지 설정을 바꿔보는데 이때 활성함수를 바꿔보는 것도 그 설정 범위에 들어간다. 그리고 인간의 뇌 연산을 최대한 모방하고 효율적인 딥러닝 모델을 만들기 위해서 활성함수를 집중적으로 연구하는 연구자도 있다. 그렇다면 활성함수 이외에는 어떤 설정이 있을까? 한 층에 들어가는 뉴런(신경)의 개수, 전체 레이어(층)수, 딥러닝 모델의 학습 방법 등 정말 다양하다.

전 세계의 수많은 연구자가 이런 딥러닝 모델의 개선에 많은 노력을 하고 있지만, 아직 딥러닝 모델 설계에 확실한 정답은 없기에 수많은 설정을 조정하면서 모델을 만들어야 하는 고충이 있다. 즉, 하나의 훌륭한 결과물을 얻기 위해서는 상당히 많은 시간과 노력이 들어가니 본격적으로 딥러닝 모델을 만들기 이전에 이 점 염두에 두고 시작하길 바란다.

이제 적당히 개념도 알고 용어도 이해했다면 인공지능을 제대로 공부하고 싶다는 마음이 생길 수 있다. 이즈음되면 주변에 잔뜩 떠들고 다니는 경우가 많은데, 이렇게 인공지능을 공부한다고 하면 다양한 의견이 쏟아진다. 그중에서 대표적인 반응은 다음과 같다.

"수학부터 해야지!"
"수학 이런 거 몰라도 다 할 수 있어!"

이렇게 크게 둘로 나뉜다. 둘 다 맞는 말이긴 한데 각자의 상황이 다르므로 그대로 들으면 안 된다. 사람들은 생각보다 친절하게 조언해주지 않으며, 조언을 듣는 사람도 비판적인 시각을 가져야 한다. 그럼 나는 어떻게 인공지능을 알아가야 할까? 같은 목표에 서로 다른 두 개 이상의 상반된 의견 또는 해답이 존재한다면 그것을 나누는 명확한 기준이 존재한다는 것이다. 여기서 추가할 기준은 주어진 시간이나 구현할 인공지능 관련 산출물의 질적 수준이 되겠다.

딥러닝은 생각보다 제법 많은 수학적 지식을 기반으로 하여 동작하는 알고리즘이다. 선형대수와 미적분학이 기본으로 들어가고 기하 벡터 등 다양한 학문이 버무려져 있다. 물론 이 지식을 처음부터 차근차근 공부하는 것도 방법이 되겠다. 하지만 대부분의 사람은 밑바닥부터 시작하기에는 시간적 여유가 없기에 대부분 수학 공부를 뛰어넘고 코드와 개념 공부만 하는 경우가 많다. 이런 경우는 향후 과학적이고 체계적인 딥러닝 모델의 설계가 어려울 수 있으며 응용력이 떨어지는 단점이 있으니 이를 감안하여 공부하도록 하자.

인공지능은 그 활용 분야가 매우 많은데 논문을 기반으로 기술범위와 사례를 보고 싶다면 paperswithcode.com 사이트를 한 번 방문하는 것을 권장한다.

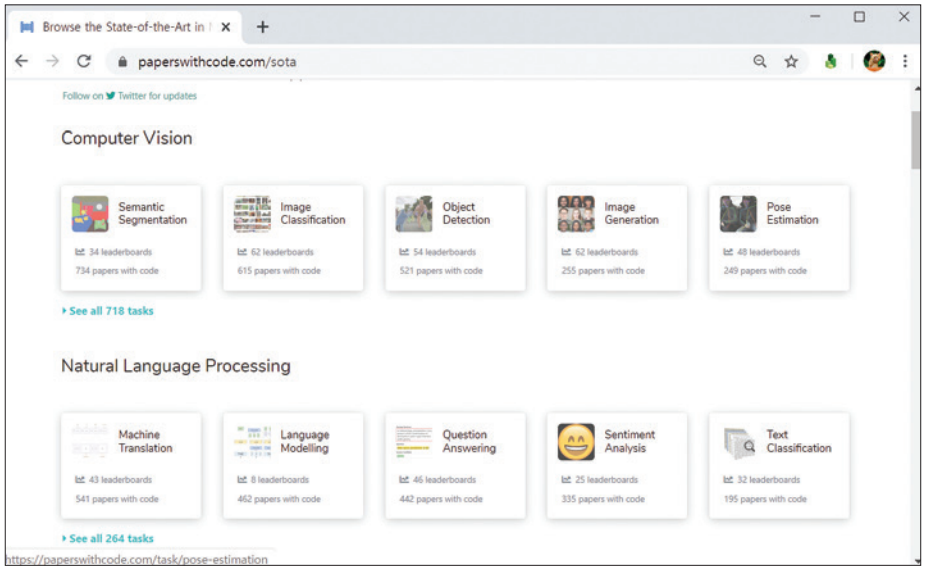


그림 12. paperswithcode.com/sota

해당 웹 페이지를 돌아다니다 보면 본인의 업무 분야와 관련 있는 내용도 있을 것이고 타 분야의 활용 사례를 보면서 또 다른 인사이트를 얻을 수 있지 않을까 한다. 게다가 다음에는 이번에 못다 한 인공지능 이야기와 더불어 가장 친숙한 사람의 언어. 우리말을 어떻게 처리해야 하는지 알아보도록 할 예정이니 자연어 처리(Natural Language Processing) 부분을 미리 찾아보는 것도 좋겠다.