AI, 머신러닝, 딥러닝의 성공 배경과 공사 업무 적용 가능성 탐색
(VOC 담당부서 자동 지정을 사례로)

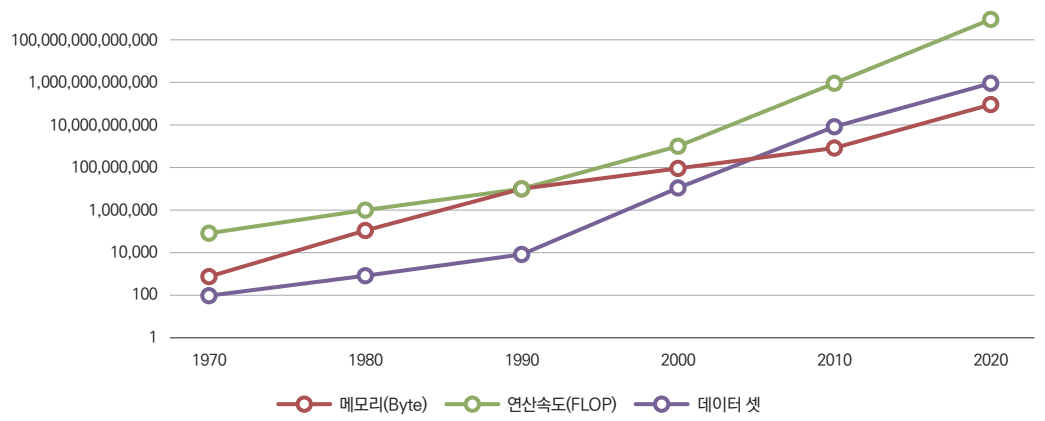
김 동 현(ICT운영부 팀장)

AI, 머신러닝, 딥러닝의 성공 배경과 개인 및 기업의 AI 역량 강화 방법을 살펴보고, 공사 업무 적용 가능성을 VOC 담당부서 자동 지정을 사례로 탐색해 보고자 함

머신러닝은 알고리즘을 이용해 데이터를 분석 및 학습하여, 학습한 내용을 기반으로 판단이나 예측을 하는 AI의 한 분야이며, 딥러닝은 여러 층의 알고리즘을 통해 데이터의 학습 및 예측 능력을 획기적으로 향상시킨 머신러닝의 한 분야임

AI, 머신러닝, 딥러닝의 성공 배경

〈기하급수적으로 성장하는 컴퓨팅 성능〉



※ Dive to Deep Learning, https://d2l.ai

컴퓨터 연산 속도, 메모리 저장 능력의 기하급수적인 증가

- 딥러닝은 신경망 네트워크(neural network)라는 이름으로 1960대부터 존재했으나, 컴퓨터 처리 능력의 한계로 인해 실용적인 활용이 불가능했음
- 20년 전에 1년이 걸리는 연산이 지금은 1시간만에 끝날 정도로 컴퓨터 성능이 기하급수적으로 향상되었으며, 지금은 클라우드에서 필요한 서버를 빠르게 생성해 딥러닝 모델 학습 및 추론이 가능함
- 위 그래프를 보면 연산속도 향상에 비해 메모리 저장 능력 향상이 더딘 것을 알 수 있음. ChatGPT-4의 경우 조 단위의 파라미터를 사용하고 있는 것으로 알려졌으며, AI 성능 향상을 위해 메모리 기술 발전 필요

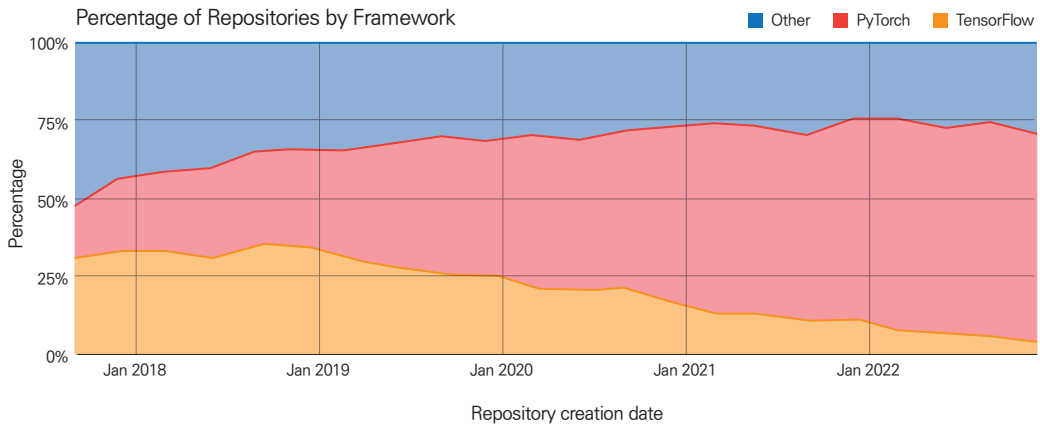
소셜 네트워크, 센서 등에서 생성되는 방대한 학습 가능 데이터

- 인터넷을 통해 대량의 학습 데이터 취득이 가능해짐. 위키피디아, 소셜네트워크, 도서로 이루어진 양질의 말뭉치 데이터 존재 덕분에 ChatGPT 같은 대규모 언어 모델(LLM; Large Languate Model) 개발이 가능하게 됨

파이토치(PyTorch), 텐서플로우(TensorFlow) 등 오픈소스 기반의 머신러닝 프레임워크 활성화

- 오픈소스는 다수의 전문가가 자발적인 참여로 만들어지는 공개 소프트웨어이며, 머신러닝 오픈소스를 이용해 복잡한 머신러닝 알고리즘을 누구나 쉽게 활용 및 개발할 수 있게 됨

〈머신러닝 프레임워크별 이용 트렌드〉



※ AssemblyAI

- 파이토치는 파이썬 언어와 유사한 간결한 문법, 다른 파이썬 라이브러리와 잘 호환되는 장점 때문에 리서치 커뮤니티에서 가장 많이 사용되는 머신러닝 프레임워크로 자리 잡음
- ChatGPT와 유사한 성능의 LLM 모델인 LLaMA를 Meta에서 2023년 2월 오픈소스로 공개하였며, Alpaca, Vicuna 등 오픈소스 LLM 모델이 개발되고 있음. 앞으로 빅테크 기업

위주의 AI 개발 보다는 오픈소스 커뮤니티의 협업 형태의 개발이 활발히 이루어 질 것으로 보임

■ 어텐션 등 획기적인 머신러닝 알고리즘 개발

- 인공 신경망이 다양한 내용이 담긴 긴 문장이나 복잡한 영상을 효과적으로 학습하지 못하는 문제가 있었으나, 전체 데이터 중 집중해야 할 부분을 찾아 학습하는 방법인 어텐션 메커니즘(Attention Mechanism)이 개발되어 자연어 처리, 영상처리 성능이 획기적으로 발전했음

AI, 머신러닝, 딥러닝 적용 사례

■ 카카오 모빌리티 – 텐서플로우*를 사용하여 차량 배차 시스템에 적용

■ 네이버 쇼핑 – 텐서플로우를 활용하여 매일 2천만 건 이상의 상품 자동 분류에 적용

■ 당근마켓 – TFX*를 사용하여 게시물 신고 업무 자동화

* 텐서플로우, TFX : 머신러닝 프레임워크

■ AI, 머신러닝 적용이 시도되고 있는 분야

- 신약 개발 / 자율 주행 / 해킹, 보안 / 금융시장 트레이딩 / LLM을 활용한 법률 서비스

AI, 머신러닝, 딥러닝 역량 강화 방법

■ 코딩을 이용한 데이터 처리능력

- 기업에서 데이터 사이언티스트가 데이터 처리를 전문적으로 수행했던 과거와는 달리 지금은 코딩을 활용한 데이터 처리 능력이 필수 소양이 됨
- 데이터 분석에 파이썬 프로그램 언어가 많이 사용됨. 파이썬은 데이터 처리, 수학적 알고리즘 표현을 쉽고 간단하게 할 수 있어 머신러닝 알고리즘 구현에 많이 이용되며, 많은 머신러닝 프레임워크의 기반 언어로 사용되고 있음
- 주로 파이썬을 이용해 머신러닝 알고리즘을 연구 및 개발하고, C, C++ 등으로 변환해 성능을 최적화하여 운영 환경에서 사용됨

■ 수학적 분석 능력

- 수학은 머신러닝 알고리즘을 이론적으로 표현하는 언어임. 머신러닝 알고리즘을 분석하고 개선하기 위해서는 선형대수, 미적분학, 확률/통계 분야의 기초적인 역량 필요

■ 오픈소스 커뮤니티 참여 및 활용

- 기업, 개인이 AI 모델을 백지 상태에서 구축하는 것은 현실적으로 불가능함. 오픈소스로

공개되어 있는 높은 품질의 최신 기술을 습득, 활용하여 AI 모델 개발 및 이용 필요. Hugging Face, OpenCV, fast.ai 등 자연어 처리, 영상 처리와 같은 다양한 분야의 AI 오픈소스 커뮤니티가 있음

■ 먼저 간단한 것부터 머신러닝 적용 시도해 보기

- 공사 업무 중 머신러닝을 바로 시도해 볼 수 있는 분야 – VOC 담당부서 자동 지정, 홈즈 주택 수 검증 등

머신러닝 모델을 이용해 공사 VOC 담당부서 분류 자동화하기 (공사 업무에 머신러닝 적용 사례)

■ 연간 3만 건이 넘는 VOC(고객의 소리)가 공사 홈페이지를 통해 접수되고 있으며 고객만족부 VOC 담당자가 VOC 내용에 따라 담당부서를 지정하고 있음. 2020년부터 축적된 VOC 담당부서 분류 결과를 머신러닝을 이용해 학습하여 담당부서 지정 업무를 자동화 가능성을 탐색해 봄

- 학습 데이터: 2020년 1월 ~ 2023년 2월에 등록된 83,863건의 VOC(고객의 소리) 중 등록 건 수가 10건 이하인 부서와 부서 분류 없이 콜센터에서 직접 처리한 건을 제외한 59,059건을 사용
- 검증 데이터: 2023년 3월 ~ 5월에 등록된 VOC 3,695건

■ 개발 방법

- VOC 텍스트를 벡터로 변환하기 위해 먼저 VOC에서 사용된 단어 중 상위 40,000개로 단어 집합을 구성

〈VOC 단어 집합〉

단어	사용 빈도
대출	51,998
보금자리론	44,947
현재	35,745
…	…

- 단어 집합을 이용한 원-핫 인코딩(One-Hot Encoding) 방식으로 각각의 VOC 텍스트를 하나의 벡터 로 변환. 원-핫 인코딩은 단어의 유사도나 문맥을 표현하지 못하나 구현이 쉽고 빠르게 동작한다는 장점이 있음

〈VOC 텍스트 원-핫 인코딩 예시〉

5월에 신청한 보금자리론 대출 실행 예정인데 … → = [1, 1, 0, …, 0, 0, 0]

- 머신러닝 프레임워크 파이토치(PyTorch)의 Linear 모델을 이용해 아래 수식의 다층 퍼셉트론(Multi-Layer Perceptron; MLP)을 구현

$$h = relu(W_1^T + b_1)$$
$$y = hW_2^T + b_2$$

〈파이토치 코드로 표현한 머신러닝 모델〉

```
class NeuralNetwork(nn.Module):
    def __init__(self):
        super(NeuralNetwork, self).__init__()
        self.linear_relu_stack = nn.Sequential(
            nn.LazyLinear(200),
            nn.ReLU(),
            nn.Dropout(p=0.5),
            nn.LazyLinear(len(brcdLst)),
        )
```

※ Github, https://github.com/dhkim9549/ai-study/blob/main/voc

- W_1, b_1, W_2, b_2는 초기에 무작위로 세팅되며 학습을 통해 추론 정확도를 높이는 방향으로 갱신됨
- 총 14개 부서를 대상으로 VOC 담당부서를 분류하며 모델의 추론을 통해 아래와 같은 결과값이 출력됨

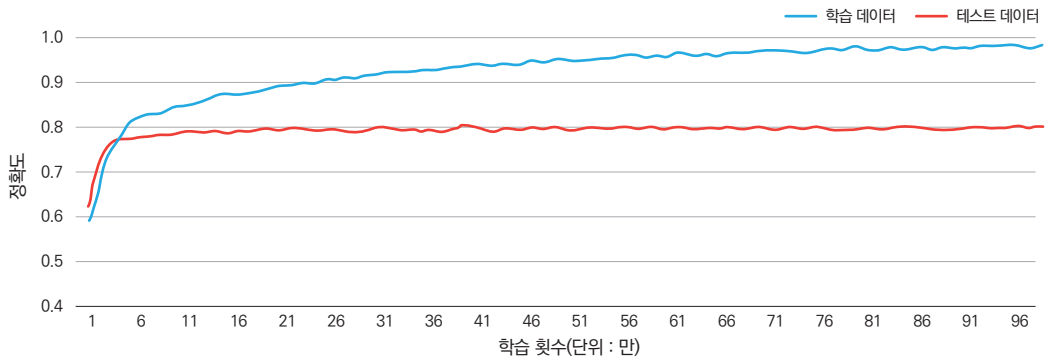
  y = [16.3155, −1.4755, −0.7394, 3.7253, 0.8227, −2.4653, −0.5528, −1.2256, −1.6660, −2.3843, −2.1420, −2.3096, −2.6978, −3.2766]

- y 벡터의 첫 번째 항목의 값이 가장 크므로 첫 번째 부서가 추론된 것으로 해석

  5월에 신청한 보금자리론 대출 실행 예정인데 … → = [1, 1, 0, …, 0, 0, 0]
  → 머신러닝 모델 → y = [16.3155, −1.4755, −0.7394, …] → 정책모기지부

적용 결과

〈학습 횟수에 따른 모델의 추론 정확도〉



- 학습 데이터에 대한 모델의 추론 정확도는 모델의 과적합(overfitting)의 결과 100%에 근접하나 검증 데이터의 경우 정확도가 80%로 나타남

〈검증 데이터의 예측 결과〉

실제 부서 \ 예측 부서	정책모기지부	지사	유동화자산부	주택보증부	주택연금부	ICT 운영부
정책모기지부	2036	87	51	18	2	0
지사	210	297	45	39	4	0
유동화자산부	122	23	196	2	0	0
주택보증부	24	4	9	366	1	0
주택연금부	6	8	0	0	40	0
ICT 운영부	10	1	0	0	0	15
종합금융센터	24	12	1	0	0	0
유동화증권부	2	0	0	0	0	0
사업자보증부	2	1	0	4	0	0
채권관리부	7	0	0	18	0	0
인사부	1	1	0	0	0	0
경영혁신부	3	0	0	1	0	0
홍보실	0	0	0	0	0	0
주택금융연구원	2	0	0	0	0	0

※ 종합금융센터 등 8개 부서는 해당 부서로 예측된 데이터가 없어서 표에서 생략함

■ 100만 회 학습 후 3,695건 검증 데이터에 대한 추론 정확도는 약 80%이었으며 타 사업부서에 비해 지사, 유동화자산부, 종합금융센터의 정확도가 낮은 것으로 나타났음. 이는 실제 담당부서 지정 시 VOC에 등록된 고객의 연락처로 고객의 공사 상품 이용중 여부, 관할지사 등을 조회하여 참고하기 때문인 것으로 보임

■ 모델의 예측 정확도를 개선하기 위한 향후 과제

- (멀티모달 사용) 모델 입력 값에 고객이 이용 중인 공사 상품 종류, 관할지사 등 추가
- (인코딩 방법 개선) BERT* 등 단어의 문맥 정보를 표현할 수 있는 텍스트 인코딩 방법 사용

* Bidirectional Encoder Representations from Transfomers(BERT): 2018년 구글에서 개발한 언어 모델

- (알고리즘 개선) 트렌스포머, 어텐션 등 문맥 정보를 학습하는 머신러닝 모델 사용

■ 참고 링크

- VOC 담당부서 자동 분류 프로토타입 웹페이지
  http://bada.ai/ai/voc-classification.html
- VOC 담당부서 자동 분류 모델 소스 코드
  https://github.com/dhkim9549/ai-study/tree/main/voc

참고문헌

- Zhang, Aston and Lipton, Zachary C. and Li, Mu and Smola, Alexander J., Dive into Deep Learning, 2023, https://d2l.ai/
- Simon J.D. Prince, Understanding Deep Learning, MIT Press, 2023, https://udlbook.github.io/udlbook/
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, Attention is all you need, 2017., arXiv:1706.03762
- Dylan Patel and Afzal Ahmad, Google "We Have No Moat, And Neither Does OpenAI", 2023, https://www.semianalysis.com/p/google-we-have-no-moat-and-neither
- Ryan O'Connor, PyTorch vs TensorFlow in 2023, https://www.assemblyai.com/blog/pytorch-vs-tensorflow-in-2023