Chapter 1. Intro

Myths(이에 반해 이책을 통해 이야기 하려고 하는)

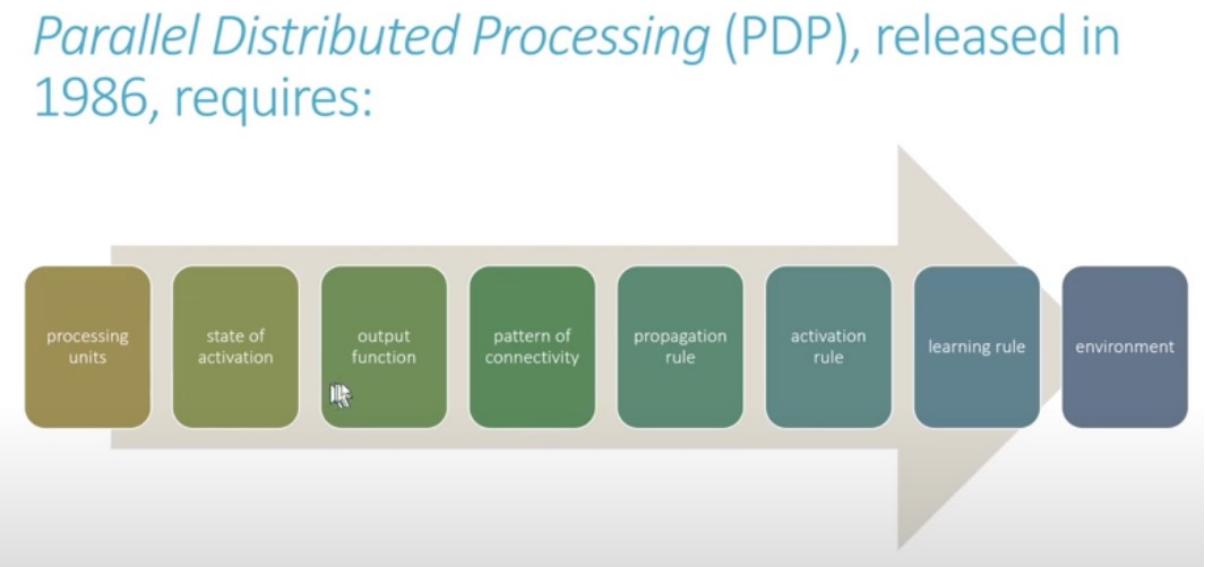
- 당신은 딥러닝이 필요없다
- 많은 수학지식과 데이터, 고성능의 하드웨어, but (반론) 우리 경험치로는 고등학교 정도 수학, 50개 미만의 데이터, 쉽게 무료로 이용할 수 있는 하드웨어 있음

Deep learning

- 적용범위 : 음성인식에서 이미지 분류에 이르기까지
- 여러 겹의 인공신경망으로 구성, 앞 단의 레이어로 부터 인풋을 받고, 각 레이어는 각기 특정 task 수행, 각 레이어는 알고리즘(에러를 줄이고 정확도를 높이는)을 통해 훈련됨
- 저자들의 생각: 더 많은 분야에 적용되어야 함(DL의 강점 power, flexibility, simplicity)
- 적용 분야 : NLP / Visions / 의학 / 생물학 / 이미지 생성 / 추천시스템 / 게임 / 로보틱스 / 기타, 인공신경망이라는 하나의 모델에 기초함

Some history

- Rosenblatt : ~ a machine capable of perceiving, recognizing and identifying its surroundings without any human training or control
- Rumelhart : Parallel Distributed Processing(PDP), 현대 인공신경망의 requirements와 유사



Deep learning 학습법(이 책에서 제시하는)

- **자신의 모델**을 더 좋게 개선하고자 하는 Motivation이 중요, 이 이후로 관련된 이론도 학습하면됨(**learning by doing: to do things before why they work)**
- E2E 전체 과정을 알고, 직관적으로 이해할 수 있는 실사례를 중심으로(향후에 수학적 이론 포함)
- 단순화 하고, 모든 사람들이 향유할 수 있게

Your projects & mindset

- 실생활에서 당신이 마주하는 문제들
- Remember : 딥러닝 분야에서 우수한 사람들이 나타내는 중요한 특질 **Playfulness & Curiosity**

Retrospect

- Liked: " AI is Everywhere " 시대 → Deep learning은 특정인의 전유물도 아니고, 통용되고 있던 진입장벽들(Myths) 없애고 대중화에 기여하고자 하는 저자들의 철학!
- Learned: Parallel Distributed Processing(PDP)의 주요 개념
- Lacked: N/A
- Longed for: 책 저자들의 대중화 노력(MOOC, 교육 등)처럼, Fastai FB 그룹이 활동해야 하는가?

Software

- PyTorch : Speed와 Simplicity 모두 충족. but, 어느 Libraries를 사용하더라고 중요치 않음
- 새로운 것들이 너무 빠르게 등장하고 있는 상황에서 정말 중요한 포인트는 밑바탕에 있는 테크닉과 어떻게 그것들을 실전에 적용하며, 새로운 툴과 테크닉들이 발표될 때 얼마나 빠르게 전문성을 키울 수 있는가에 초점을 맞추는 것임
- Jupyter notebook : 딥러닝을 학습하는데 중요한 코드 작성과 실습, 쉽게 가능케 하는 플랫폼

Your first model

- 그 모델이 왜 작동하는지 설명하기 전에 우선 그 모델을 실험해보는 approach 견지

Getting a GPU deep learning server

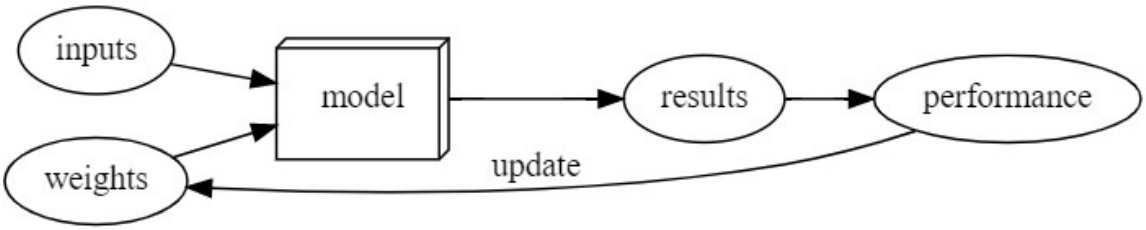
- 책 뒤에 제시된 Cloud 기반 플랫폼 사업자 선택, 초기 setup 등 번거로움 피하기

Running your first notebook

- jupyter cell : markdown 포함 셀 / 코드 포함 실행되는 셀
- jupyter model : command / edit
- 몇 가지 단축키들
- 1st Model : 개와 고양이 분류 모델(Oxford IIIT Pet dataset)
    - code 실행은 별도 파일로

What is machine learning

- 딥러닝은 인공신경망으로 구성, 머신러닝의 한 부분
    - 기존 방식은 사람이 개와 고양이 분류 step들을 다 순서에 맞게 프로그램밍해야. 그런데 사진 속에서 물체를 인식하는 순서를 어떻게 설명할 수 있는가?
    - Arthur Samuel : computer에게 step을 알려주는게 아니라, 그 문제를 어떻게 풀 수 있는지 실제 샘플을 보여 주고 학습시키는게 더욱 효과적(a machine so programmed would "learn" from its experience)

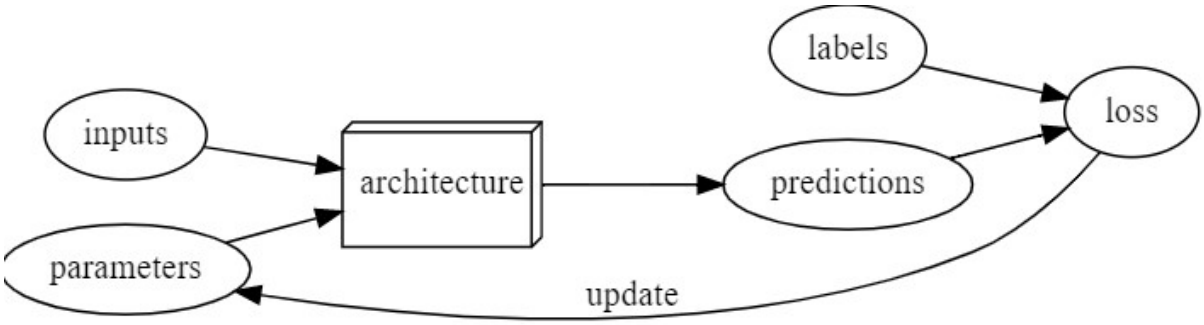


    - 4가지 주요 개념
        - weight assignment : model parameters
        - an automatic means of testing the effectiveness of any current weight assignment in terms of actual performance : 복수의 모델간 성능 우열 비교
        - we need a mechanism for altering the weight assignment so as to maximize the performance
            - 체스게임 모델에서 results는 게임 중에 말을 움직이는 행위를 performance는 게임의 승패, 얼마나 빨리 승리했는지 등 다른 의미

- 모델이 훈련되었다고 하는 의미는, 최적의 weight assignment가 이루어짐 의미

What is neural network

- 특별한 종류의 머신러닝 모델, highly flexible(다양한 분야 문제 적용) and powerful(SGD)
- function(수학적 증명) 그 자체 보다는 neural network을 훈련시키는 프로세스 즉, 좋은 weight assignments를 찾는거에 집중해야
- process : weight을 자동적으로 업데이트해주는, 그것도 일려진 SGD(경사하강법) 있음



Limitations inherent to ML

- 모델은 데이터가 없으면 만들어 지지 못함
- 모델은 훈련에 사용된 인풋 데이터의 패턴에 의존해 학습이 이루어짐
- 이 모델의 학습 어프로치는 예측값만을 반환하고 바람직한 행동을 recommend하지는 못함
  - e-commerce 추천시스템의 경우, 고객의 과거 구매 data 기반, 구매했었거나/둘러본 상품 카테고리들 중에서 추전. 고객들이 들었을 때 흥미를 가질만한 전혀 새로운 상품 추천이 아님
- 인풋 데이터 뿐 아니라 각 데이터의 라벨 또한 쌍(pair)으로 있어야 함(Labelling approach)
- 모델과 환경간의 상호작용, Feedback loop을 형성함
  - 과거 체포(arrests)가 많았던 지역 데이터를 기반으로 한 범죄 예측 모델링 : not predict crime but arrests → 언급된 특정 지역 순찰 행위 증가 → arrests 증가, but 모델 업데이트/ retrain으로 다른 결과치
  - 비디오 추천시스템 역시 유사한 편향된 결과치 강화할 위험

How our image recognizer works

- code에 대한 부연설명은 colab 파일에 마크다운 주석으로 기입
- validation set의 accuracy 중요: 긴시간 학습 → trainset 암기 → validset에 일반화 적용 어려움(성능 저하) Overfitting
- loss(measure of performance, 자동화된 weight assignment로서 SGD 방법) vs. metric(인간의 가정 사항에 기초해서 정의, 모델로 무엇을 하고자 하는지)
- pre-trained 모델의 중요성
- transfer laearning : 아직 많이 연구되지 못하고 적용 영역도 제한적(원래 학습된 목적의 영역이 아닌 다른 task에 이 학습된 모델을 적용하는 것)
- fine_tune(pre-trained 모델 기반으로 head layer만 randomly fit하는 경우) vs. fit (a model)

What our image recognizer learned

- Matt Zeile, layer가 5개인 알렉스넷을 대상으로 각각의 layer들이 training data로부터 각 feature들을 추출해 내는지 visualizing(edge → high level semantic 요소들)

Image recognizer can tackle non-image tasks

- sound를 spectrum, 주파수로 변환해 각 소리가 가지는 파형을 이미지화, 구분
- 시계열(time-series)데이터 역시 이미지 적용 가능
- 사기거래 탐지(fraud detection) : 마우스의 움직임(포지션/스피드/마우스 포인터의 움직임 가속 정도 등) 클릭을 이미지화
- 바이러스 binary 파일을 8 bit vector화해서 gray scale 이미지로 전환, 구분
- 가지고 있는 데이터를 어떻게 represent할 건지, creative한 접근이 필요

Jargon recap(일반적인 ML에 적용되는 concepts summary)

- ML은 데이터 기반의 학습에 기초한 정의된 모델(프로그램)의 한 분야이며 DL은 멀티레이어를 가진 인공신경망을 활용한 ML의 특수 분야
- 모델은 어떤 Architecture를 선택하느냐로 부터 출발. 모델 훈련 과정은 파라메터인 weights을 잘 찾는 과정으로, weights은 우리가 가진 데이터가 잘 동작하는 모델의 일반 architecure를 특정함
- 예측이 잘 작동하는 모델을 설계하기 위해 loss function을 정의함(이것이 예측의 점수를 결정함)

- 학습속도를 높이기 위해 pre-trained model을 활용. pre-trained model의 header 부분만 사용자 데이터에 맞추어 학습시키는 방식이 fine-tunning
- 모델 훈련에 있어 핵심은 모델의 일반화(새로운 dataset에서도 정확한 예측을 하는지) 역량임. 훈련과정에만 최적화된 것을 overfitting이라고 하며 이를 방지하기 위해 dataset을 train과 validation set으로 구분. 사람이 이해하기 쉽게 모델이 얼마나 성능이 좋은지(validation set을 대상으로) 정의하기 위해 만든 지표가 metric이며, 훈련 dataset을 한 cycle 도는 것을 epoch이라함
- 어떤 요소가 딥러닝을 차별적으로 만드는가라는 측면에서 보면 architecture가 중요

Deep learning is not just for image classification

- 이미지 segmentation : 자율 자동차 분야 응용(보행자 및 물체 인식 등)
- 자연어 처리(NLP) : text 생성, 자동번역, 문장 분석(영화 리뷰에 대한 sentiment analysis 등) 응용
- Tabular data : table 형태의 data로, 컬럼 feature 정보를 바탕으로 특정 컬럼 예측
  - category vs. continuous data 속성
  - tabular data의 경우 활용 가능한 pre-trained model이 거의 없는 관계로 fine-tunning이 아닌 fit_one_cycle로 학습
  - demographic data(학력/결혼유무/인종/성별)를 기초로 개인의 년간 소득이 $50K를 넘을지 예측하는 모델(80% 정확도, 30초 정도 학습시간 소요)
  - 추천시스템 : 아마존, 넷플릭스(사람들이 선호할만한 영화 타이틀 0.5~5 scale로 추천, 평균 에러 0.6. 수치 데이터 예측이므로 y_range 파라미터 사용, pre-trained model이 없지만 여기서는 fine_tuning 사용했는데, why it works는 뒤에서 다룰 예정)

Dataset

- 보유한 data의 subset(cut-down)으로 실험/프로토타이핑 후에 full-size 활용

validation sets & test sets

- training data에 대한 overfitting 문제가 있는거 처럼
- hyper-parameter(architecture, learning rates , data augmentation, 다른 요소들)를 바꾸면서 다양한 modeling을 하는 과정에서 validation set도 일정 부분 노출돼서 overfitting의 개연성 있음. 그래서 완전 평가 목적으로만 hidden 상태의 test set 필요

Use judgement in defining test sets

- valid/test dataset 선정의 핵심 고려요인은 앞으로 나타날 data의 특징을 잘 representative하는지 여부임
- time series data의 경우, randomly data를 분리하는 거는 의미 없으며, 초반부 data를 가지고 training을 하고 예측하려고 시기와 근접한 시점의 data를 validation set으로 사용해야함
- 훈련에 사용한 data와 test에 사용할 데이터가 qualitatively 다를 경우

Retrospect

- ML/DL의 중요 개념, Fastai가 적용 가능한 영역 quick review 등 개괄적으로 소개한 서론
- 실전, 실생활에서 부딪히는 문제를 풀기 위해 , 코딩과 실험이 이론적 이해보다 훨씬 중요하다는 Practical한 Approach 및 저자들의 대중화 철학과 노력에 동의 [EOD]