AI - Homework01

#AI, #용어정리

Deep Learning 기초

Deep Learning 개요

- MLP란?

MLP(MultiLayer Perceptron)은 여러 개의 Perceptron 뉴런을 여러 층으로 쌓은 다층신경망 구조이며 미리 에측이 가능한 인공적인 뉴럴 네트워크이다.

MLP는 입력층과 출력층 사이에 하나 이상의 은닉층을 가지고 있는 신경망이다.

- 합성곱 연산

합성곱 연산: CNN 신경망의 핵심으로 이미지 처리에 사용되는 연산이다.

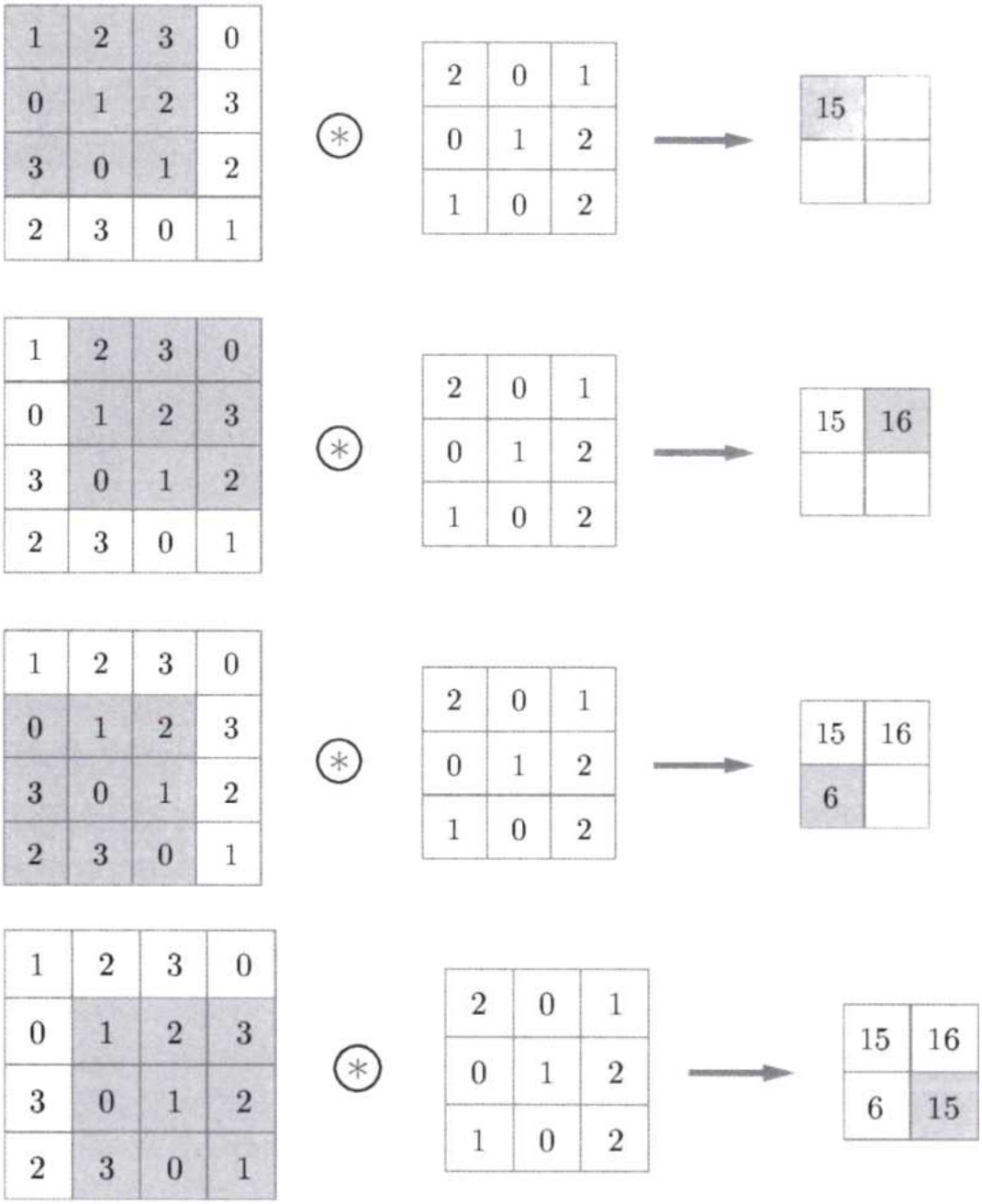
더하기와 곱만을 사용한다.

이미지에 필터를 씌우는 과정하고 똑같다.

합성곱 과정

배열을 준비하고 배열은 0부터 255까지 존재하고 255에 가까울수록 밝다.

겹치는 숫자들만 곱해서 다 더하면 된다. 왼쪽 위 부터 오른쪽으로 가고 더 이상 갈 수 없으면 왼쪽 밑으로 갔다가 오른쪽으로 가면 된다.


1 2 3 0
0 1 2 3
3 0 1 2
2 3 0 1
2 0 1
0 1 2
1 0 2
15
1 2 3 0
0 1 2 3
3 0 1 2
2 3 0 1
2 0 1
0 1 2
1 0 2
15 16
1 2 3 0
0 1 2 3
3 0 1 2
2 3 0 1
2 0 1
0 1 2
1 0 2
15 16
6
1 2 3 0
0 1 2 3
3 0 1 2
2 3 0 1
2 0 1
0 1 2
1 0 2
15 16
6 15

- 비용함수란?, 손실함수란?

통계학, 경제학 등에서 널리 쓰이는 함수로 머신러닝에서도 손실함수는 예측값과 실제값에 대한 오차를 줄이는 데에 유용하게 사용한다.

비용함수(Cost Function): 모든 오차를 일반적으로 최소화하기 위해 정의되는 함수
손실함수(Loss Function): 한 개의 데이터 포인트에서 나온 오차를 최소화하기 위해 정의 되는 함수

- 과적합이란?

기계학습에서 학습 데이터를 과하게 학습하는 것을 말한다.

알고리즘이 학습 데이터에 과하게 접합한 상태가 되어 모델이 정확한

예측이나 결론을 내릴 수 없는 경우에 발생한다.

- 방지 방법

데이터의 양 늘리기

데이터의 양을 늘릴 수록 모델은 데이터의 일반적인 패턴을 학습하여 과적합을 방지할 수 있다.

2. 모델의 복잡도 줄이기

인공 신경망의 복잡도는 은닉층의 수나 매개변수의 수 등으로 결정된다.

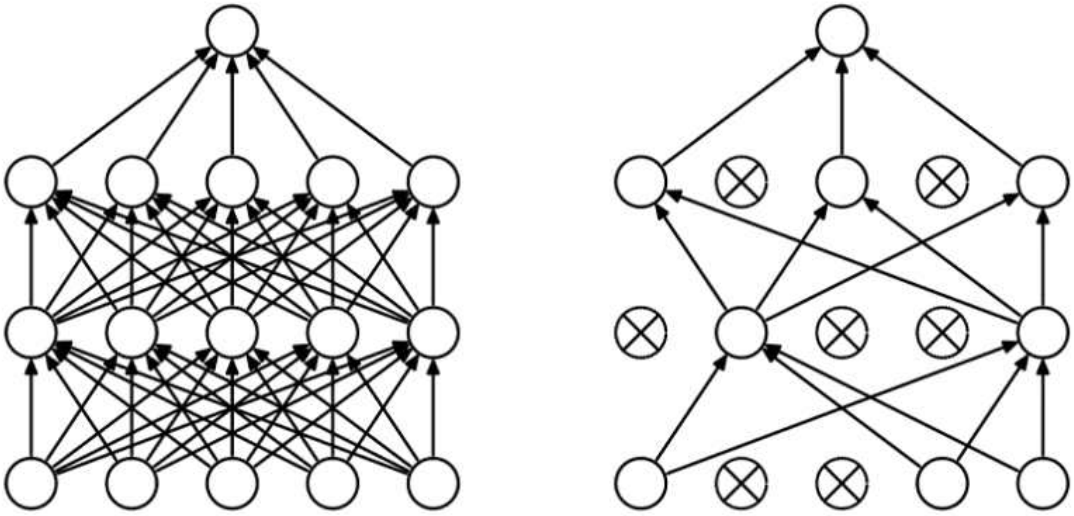
3. 가중치 규제(Regularization)적용하기

간단한 모델(적은 수의 매개변수)로 바꾸면 과적합을 예방할 수 있다.

4. 드롭아웃(Dropout)

드롭아웃은 학습 과정에서 신경망의 일부를 사용하지 않는 방법이다.

학습 시에 인공 신경망이 특정 뉴런 또는 특정 조합에 너무 의존적이게 되는 것을 방지해주고, 매번 랜덤 선택으로 뉴런들을 사용하지 않으므로 서로 다른 신경망들을 앙상블하여 사용하는 것 같은 효과를 내어 과적합을 방지한다.



(a) Standard Neural Net (b) After applying dropout.

5. 배치 정규화(Batch Normalization)

뉴럴넷에서 각 *활성함수의 미분값은 역전파 과정에서 계속 곱해지기 때문에 굉장히 중요하다. *시그모이드 함수의 경우 일정수준 이상 혹은 이하의 값이 입력되었을 때 미분값이 0에 가깝게 되는데, 이때 파라미터 업데이트 과정에서 0에 가까운 값이 지속적으로 곱해지면 기울기 소실 문제가 발생한다. 그러면 파라미터 업데이트가 거의 일어나지않고 수렴 속도도 아주 느리게 되어 최적화에 실패하게 된다.

*활성함수: 신경망 회로에서, 한 노드에 대해 입력값을 다음 노드에 보낼지 말지에 대해 결정하는 함수

*시그모이드 함수: 신경망에서 자주 사용하는 활성화 함수

6. 교차 검증(Cross-validation)

모델의 학습 과정에서 학습 / 검증데이터를 나눌때 단순히 1번 나누는게 아니라 k번 나눈 뒤

모델의 학습을 검증하는 방식이다.

예를 들어 데이터 하나가 있다.

데이터에는 train set, test set으로 구성되어 있다. 고정된 test set을 가지고 모델의 성능을 확인하고 파라미터를 수정하고, 이 과정을 반복하면 결국 내가 만든 모델은 test set에만 잘 동작하는 모델이 된다. 이 경우에는 과적합이 되어 다른 실제 데이터를 가지고 예측을 수행하면 엉망인 결과가 나와버리게 된다. 그러니 학습 / 검증데이터를 나눌때 k번 나눠야 과적합이 발생 하지않는다. 하지만 교차 검증은 모델 훈련/ 평가 시간이 오래 걸린다.

pytorch 다루기

- pytorch 란?

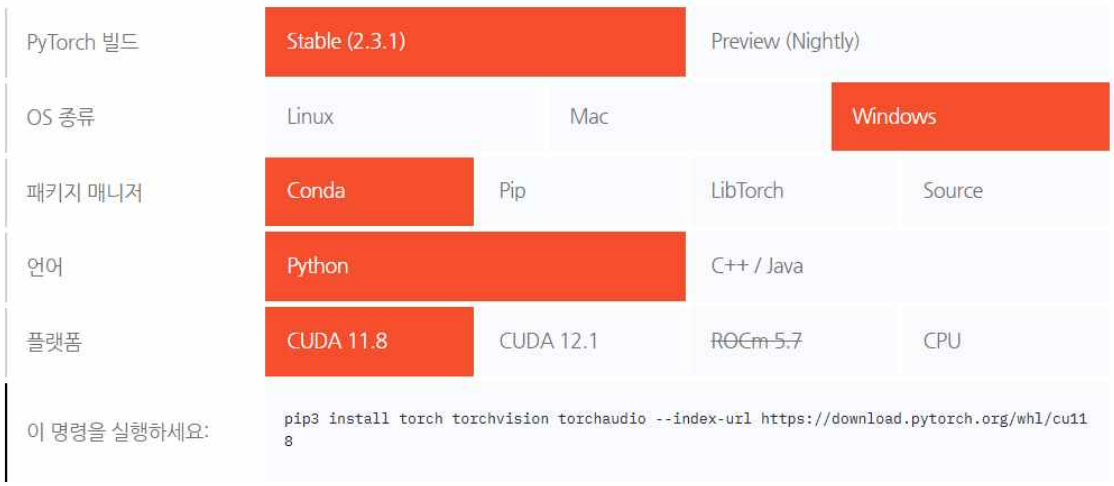
인공지능 연구 및 개발을 위한 신경망 구축에 사용되는 소프트웨어 기반 오픈소스 머신러닝 프레임워크이다.

Torch의 머신 러닝 라이브러리와 Python 기반의 고급 API를 결합한 것이다.

- 환경설정(설치 방법)

파이토치 사이트 접속 및 커맨드 입력

파이토치 공식 사이트인 https://pytorch.org/에 접속



위와 같은 설정에서 각 항목을 아래 기준으로 선택한다.

PyTorch Build : 현재 가장 안정 버전 / 최신 베타 버전 / 장기 지원 예전 버전 중 선택

Your OS : 리눅스 / 맥 / 윈도우 중 사용 운영체제 선택

Package : 가상환경이라면 Conda, 로컬환경이라면 Conda / Pip 중 선택 권장

Language : 사용할 언어 환경 선택

Compute Platform : GPU가 있다면 CUDA 버전에 맞게 선택, GPU가 없다면 CPU 선택

만일 CUDA 버전이 10.1이라면 =10.2 부분 커맨드를 =10.1 식으로 고쳐서 사용

이후 아래의 커맨드 메시지를 복사하여 터미널에 입력하거나

주피터 노트북에서 !를 앞에 붙여서 입력한 뒤 실행하기

2. 환경 체크 및 GPU 확인

환경마다 다르지만 커맨드 입력 후 설치에는 수 분 가량이 소요될 수 있다.

설치 후, 파이토치 import 및 GPU 확인이 성공적으로 된다면 설치가 완료된 것임.

(CPU 버전으로 설치한 경우에는 GPU 확인 결과가 False로 나온다.)

pytorch 구성요소, 문법, 특징

pytorch의 구성요소

torch: 메인 네임스페이스, *텐서 등의 다양한 수학 함수가 포함

torch.autograd: 자동 미분 기능을 제공하는 라이브러리

torch.nn: 신경망 구축을 위한 데이터 구조나 레이어 등의 라이브러리

torch.multiprocessing: 병렬처리 기능을 제공하는 라이브러리

torch.optim: SGD(Stochastic Gredient Descent)를 중심으로 한 파라미터 최적화 알고리즘 제공

torch.untils: 데이터 조작 등 유틸리티 기능 제공

torch.onnx: 서로 다른 프레임워크 간의 모델을 공유할 때 사용

*텐서: 물리학과 공학에 많이 사용되는 배열, 행렬과 매우 유사한

- pytorch의 문법

tensor 생성

-> import torch ->> pytorch의 문법이나 메소드들을 사용할 수 있다.

tensor의 종류

scalar => np.array(10) // 0차원, 1개의 데이터만 존재하므로

vector => torch.FloatTensor([0,1]) //원소 0과 1을 가진 1차원 텐서 생성

matrix => torch.FloatTensor(3,2) //임의의 원소값 가진 3행 2열의 2차원 텐서 생성

tensor => torch.FloatTensor (3,2,1) // shape을 기반으로 임의의 원소값을 가진 3차원 텐서 생성, 높이 k = 3, 너비 n = 2, 깊이 m = 1

텐서 생성의 주요 메소드

torch.zeros(x,y,z) : 0 값으로 shape 모양의 텐서 생성 (꼭 x,y,z 를 필수로 지정해주지 않아도 됨)

torch.rand(x,y,z) : 0부터 1 사이에 균일한 확률 분포로 랜덤한 값을 가진 (x,y,z) shape의 텐서 생성 = 균일한 확률 분포이므로 될 확률이 다 똑같음 .

torch.randn(x,y,z) : 표준 정규 분포를 따르는 랜덤값 가지는 (x,y,z) 텐서 생성 = 숫자마다 확률 다름

torch.full((x,y,z),a) : a값으로 모두 채우는 텐서 생성

2. tensor 변환

reshape() : 배열 구조 변경

view() : 텐서 구조 변경

데이터 타입 변환(type)

data1 = data1.type(torch.float32)

전치 행렬

print(data2)

data2 = data2.T

=> 행과 열을 교환

arange(): 1차원 tensor 생성

linspace(): 범위 내 1차원 tensor 균등 생성

3. tensor 연산

행렬 곱셈일 때, 앞 행렬의 열의 갯수와 뒷 행렬의 행의 갯수가 같아야한다.

tensor 연산은 shape 이 동일해야 하고, 행과 열이 같은 값끼리 연산이 된다.

즉, 행과 열이 같아야 행렬의 곱셈 수행 가능

pytorch의 기능

간결하고 구현이 빠르게 진행되며, TensorFlow보다 사용자가 익히기 훨씬 쉽다는 특징이 있다.

pytorch는 python 코딩과 비슷하기 때문에 언어 자제에 대한 어려움은 없으며, 선언과 동시에 데이터를 집어넣고 세션도 필요 없기 때문에, 코드가 간결하고 난이도가 낮은 편이다.

데이터 전처리

- 데이터 전처리란?

데이터 분석 및 적합한 형태로 만드는 과정을 총칭한다.

데이터 전처리는 데이터 분석 및 처리 과정에서 중요한 단계이고 데이터 분석, 데이터 마이닝, 머신 러닝 프로젝트에 적용한다.

- 데이터 전처리 목적성

원시 데이터로부터 유용한 정보를 추출하고, 머신러닝 모델이 이를 효과적으로 학습하고 일반화할 수 있도록 데이터의 품질을 향상시키는 것이다.

데이터 품질 향상

모델의 성능 향상

일반화 증진

- 데이터 전처리 절차 및 방법

데이터 전처리 절차:

데이터 수집 -> 데이터 정제 -> 데이터 통합 -> 데이터 축소 -> 데이터 변환

데이터 전처리 방법:

결측치 처리: 데이터에서 누락된 값들을 적절히 처리하는 핵심 과정

- 삭제

- 대체

- 보간

2. 이상치 처리: 모델의 학습을 왜곡하지 않도록 하는데 중요한 역할을 한다.

- 통계적 방법

- IQR 기반

3. 특성 공학: 데이터의 본질적인 특징을 잘 파악할 수 있도록 돕는 핵심 전처리 과정

- 다항 특성 추가

- 상호 작용 특성 추가

4. 스케일링: 특성 간의 크기 차이를 조정하여 모델의 성능을 향상시키는데 중요하다.

- 표준화

- 정규화

5. 일반화 증진

- 데이터 증강

- 드롭아웃

- 데이터 전처리 기초 문법

파일 오픈

3가지 명령

파일 오픈

파일 읽기 또는 쓰기

파일 닫기

encoding = 파일 오픈시, 해당 파일의 인코딩 방식을 명시

파일디스크립터변수 = open(파일이름, 파일열기모드)

절대 경로 = 최초의 시작점으로 경유한 경로를 전부 기입하는 방식

윈도우 예: 드라이브명부터 시작, C:\ 등 C:\Users\UserID\Desktop\test.txt

맥 예 : 최상단 디렉토리를 나타내는 / 부터 시작 /Users/davelee/test.txt

상대 경로 = 파일을 찾는 위치부터 상대적인 경로를 기입

상위 폴더는 ../ 로 명시할 수 있음

../testfile.txt

2. 파일 닫기

파일디스크립터 변수 .close() 함수로 파일을 닫을 수 있음

다양한 포멧의 파일이해

2-1. csv

for문을 사용하여 정렬> plain text와 마찬가지로 open 후에는 클로즈

csv reader : csv리스트로 변환, 읽기용

csv 파일을 open할때는 newline옵션(윈도우에서 오류방지)

csv writer : csv파일을 쓸 때

write row

csv 함수 대신에 DictWriter : 최상단에 csv 필드값정의

-pandas 라이브러리로 csv읽기

2-2. xml

데이터에 특정값을 부여하기위해서는 문자만으로는 표현에 한계가 있음, 구조화하는 포맷(대세는 jason)

html과 유사한 마크업하는 범용 포맷

기본구조 : html과 동일, 대신 속성 값은 임의

태그를 이용하여 그룹화 가능

태그는 무한 증식가능

xml을 분석하는 라이브러리 사용할것

파싱 (parsing)과 데이터 추출 코드 : 일종의 xml 데이터를 분석해서, 빠르게 원하는 데이터를 추출할 수 있도록 트리(tree) 형태로 만드는 것

2-3. xml 데이터 처리 순서

#1 xml 데이터 읽기

#2 xml 데이터 파싱하기

#3 원하는 데이터 관련 태그 선택하기

#4 리스트이므로 for 문으로 아이템 추출

#5 각 아이템. text로 원하는 데이터 출력

2-4. jason

javaScript Object Notation 줄임말

JSON은 서버와 클라이언트 또는 컴퓨터/프로그램 사이에 데이터를 주고 받을 때 사용하는 데이터포맷

키와 값을 간단한 기호로 구성하여 표현 가능(괄호, 세미콜론 등), 운영체제 구애안받음

앱 웹 환경에서 Reset API 사용하여 서버 클라이언트 데이터 주고 받는 경우에 많이 사용함

#1 데이터포맷읽기 : 제이슨 라이브러리 함수

jason.loads 함수를 이용하여 데이터 파싱 가능(문자열을 jason으로)

jason.dump()함수를 이용하여 데이터를 통째로 문자열로 변환가능 dump한 후에 indent를 활용하여 정렬, 파일로 쓰기 가능

jason load : 파일로된 jason 데이터를 사전처럼 변환

코랩 사용

구글 코랩이란?

구글이 제공하는 클라우드 기반 Jupyter Notebook 환경

웹 브라우저에서 파이썬 코드를 작성하고 실행할 수 있는 웹 에디터 라고 볼 수 있음

사용방법:

구글 코랩 -> 파일 새노트 -> 로그인 -> 노트이름 바꾸고 사용

*단축기

[실행 관련 단축기]

Ctrl + Enter = 해당 셀을 실행하고 커서를 해당 셀에 두는 경우(결과 값만 보고자 할 때)

Shift + Enter =해당 셀을 실행하고 커서를 다음 셀로 넘기는 경우(여러가지 값을 빠르게 출력할 때)

Alt + Enter = 해당 셀을 실행하고 셀을 삽입한 후 커서를 삽입한 셀로 넘기는 경우(다음 작업 공간이 없을 때)

[셀 삽입/삭제 관련 단축기]

Ctrl + M A = 코드 셀 위에 삽입

Ctrl + M B = 코드 셀 아래 삽입

Ctrl + M D = 셀 지우기

Ctrl + M Y = 코드 셀로 변경

Ctrl + M M = 마크다운(텍스트) 셀로 변경

Ctrl + M Z = 실행취소

<출처> 구글