



DEEP
LEARNING
INSTITUTE

(<https://www.nvidia.com/en-us/deep-learning-ai/education/>)

새로운 데이터를 처리하는 모델 만들기

빅 데이터의 시대에 우리는 하루에 ~ 2.5 쿼틸리언($=10^{18}$)의 데이터를 만들어 냅니다. 이들 데이터 중 무료 데이터 세트들은 [Kaggle.com](https://www.kaggle.com/datasets) (<https://www.kaggle.com/datasets>)이나 [UCI](https://archive.ics.uci.edu/ml/datasets.html) (<https://archive.ics.uci.edu/ml/datasets.html>) 같은 사이트에서 얻을 수 있습니다. 다양한 창의적 방법을 통해 여러 사람들이 협업하여 데이터 세트를 만듭니다. 예를 들어 페이스북은 사용자들에게 친구들의 사진을 태그하도록 하여 레이블 있는 안면 인식 데이터 세트를 만듭니다. 더 복잡한 데이터 세트는 전문가들에 의해 수동으로 만들어집니다. 가령 방사선 전문의들에게 심장의 특정 부위에 레이블을 붙여 달라는 식으로 말입니다.

지금까지 우리는 16 개의 이미지를 가진 *매우 작은 데이터 세트*를 가지고 작업했습니다. 우리는 실제로 수 만 개 정도의 레이블이 붙은 이미지를 사용해야 합니다만, 루이 이미지를 (아직은) 그만큼 가지고 있지 않습니다. 다행히도 Kaggle에는 우리가 바로 사용할 수 있는 18,750 장의 레이블 붙은 개와 고양이 이미지가 있습니다. 우리의 신경망에게 루이가 누구인지 학습시키는 대신에, 무엇이 개인지를 학습시켜 봅시다.

이 과제가 끝나면, 여러분은 학습 데이터 세트에 있지 않은 개와 고양이 이미지를 올바르게 분류하는 신경망을 가지게 됩니다.

우선은 레이블 있는 데이터로부터 학습 데이터 세트를 만들면서 DIGITS와 심층 학습 작업 흐름에 대하여 조금 더 알아보도록 하겠습니다.

[Open DIGITS \(/digits/\)](#).

우리의 첫 데이터 세트 불러오기

DIGITS를 시작하면 새로운 데이터 세트와 모델을 만들 수 있는 홈 화면으로 가게 됩니다.

왼쪽의 데이터 세트(Datasets) 탭을 클릭하여 시작하세요.

지난 강좌에서 사용한 "비글 이미지(Images of Beagles)"는 당분간 무시하세요. 우리의 신경망이 각 이미지가 속하는 "클래스"를 말해주도록 하기 위해 우리는 DIGITS에게 **새로운** "분류" 이미지 데이터 세트를 만들라고 요청할 것입니다. 오른쪽의 파란 "이미지(Images)" 메뉴에서 "분류(Classification)"를 선택하세요.

이제 사용자 이름을 입력해야 할지도 모릅니다. 입력 요청이 있으면 아무 이름이나 소문자로 입력하세요.

데이터 불러오고 조직화하기

"학습 이미지(Training Images)" 필드에 아래 디렉토리를 입력해서 DIGITS에게 데이터가 어디에 있는지 알려주세요.

```
/dli/data/dogscats/train
```

이 폴더는 여러분이 작업하는 컴퓨터가 아니라 학습 환경에 있는 폴더임에 유의하세요. 여러분의 컴퓨터와 학습 환경 사이에 데이터를 서로 전달하는 다양한 방법이 있습니다. 계속 업데이트 중인 문서 [Resources: Next Steps for Independent Deep Learning](https://docs.google.com/document/d/1A8r1Shh0ssiRzrxNcraK7PJ_NUFay--EX1aBovpVMKU/edit) (https://docs.google.com/document/d/1A8r1Shh0ssiRzrxNcraK7PJ_NUFay--EX1aBovpVMKU/edit)을 참고하세요.

데이터 세트를 불러오기 전에 데이터가 어떻게 구성되었는가를 알고 싶으시면 "학습 이미지" 옆에 있는 물음표 (?) 위에 마우스 커서를 올려 놓으세요.

입력 데이터를 출력 레이블과 짝지우는 많은 방법이 있습니다. 우리는 개 이미지가 담긴 폴더 "dogs"와 고양이 이미지 폴더인 "cats"를 만들었습니다. 이들을 다시 "dogscats"라는 폴더에 넣었습니다. DIGITS는 두 개의 폴더가 있다는 것을 알 수 있고, 총 2 개의 클래스가 있다고 인식합니다. 이것이 다입니다. 여러분은 곧 여러분의 데이터로 이것을 연습해볼 것입니다.

검증 (% for validation) 필드에 25%를 할당한 것에 유의하세요. 데이터 세트가 불러오는 중에 이것에 대하여 읽을 기회가 있을 것입니다.

데이터 세트를 준비하는 시간을 줄이기 위해 "인코딩(Encoding)"을 "PNG(무손실)"에서 "없음(None)"으로 바꾸고 데이터 세트에 이름을 지정하세요.

"생성(Create)"을 클릭하면 DIGITS는 약 3 분에 걸쳐 학습할 데이터를 준비할 것입니다. 그러는 동안에 아래 섹션을 읽으시고 어떤 일이 일어나고 있는지 알아보세요.

DB backend

LMDB

Image Encoding ?

None

Group Name

Dataset Name

Dogs and Cats

Create

DIGITS의 역할

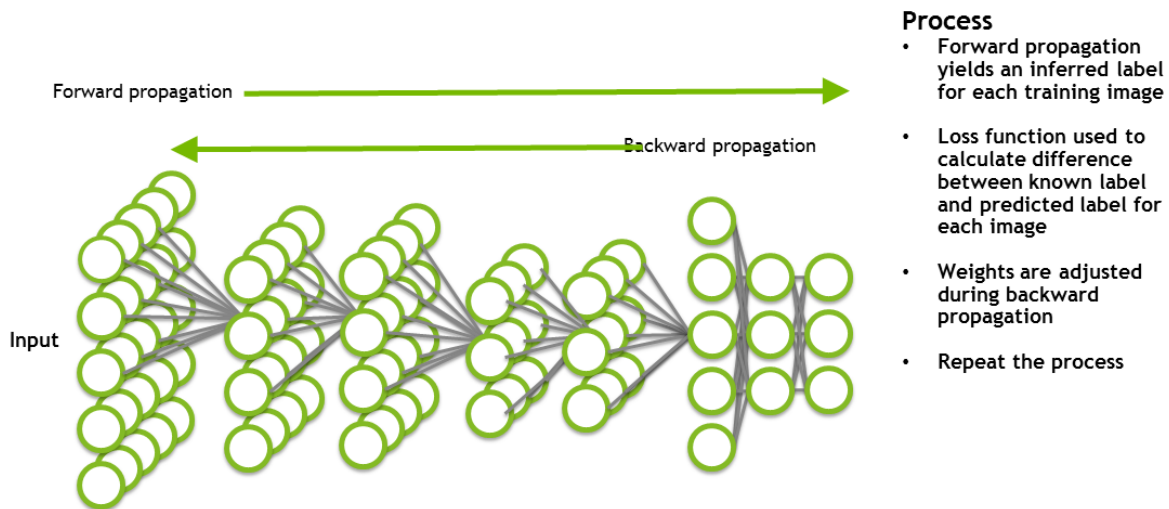
여러분은 DIGITS에게 지정된 폴더의 이미지를 가져다가 아래 두 가지 작업을 하도록 했습니다.

1) 여러분이 학습시키는 신경망이 원하는 크기와 맞도록 이미지를 같은 크기로 표준화합니다. 우리는 또다시 256x256 컬러 이미지를 입력으로 하는 AlexNet을 학습시킬 것입니다. 다음 과제에서 더 자세히 다루겠습니다.

2) 이미지를 두 개의 데이터 세트로 나눕니다. 각 클래스에서 75%의 이미지는 학습에 사용하고 25%는 검증에 사용합니다.

학습 데이터 세트는 우리가 루이 분류기를 학습시킬 때 보았던 방식으로 사용될 것입니다. 즉, *이미지를 신경망에 순전파시키고, 출력을 생성하고, 손실을 계산하고, 가중치를 갱신하기 위해 손실을 신경망에 역전파*합니다. (이들이 바로 심층 학습의 주요 요소입니다만, 여기에서 자세히 다루지는 않습니다. *컴퓨터가 수행하는 계산을 이해*해야만 심층 학습을 이용하여 문제를 풀 수 있는 것은 아니니까요. 하지만 관심 있는 분들은 [3Blue1Brown에서 제공하는 동영상](https://www.youtube.com/watch?v=llg3gGewQ5U) (<https://www.youtube.com/watch?v=llg3gGewQ5U>)을 저장해두었다가 나중에 보시면 좋습니다.)

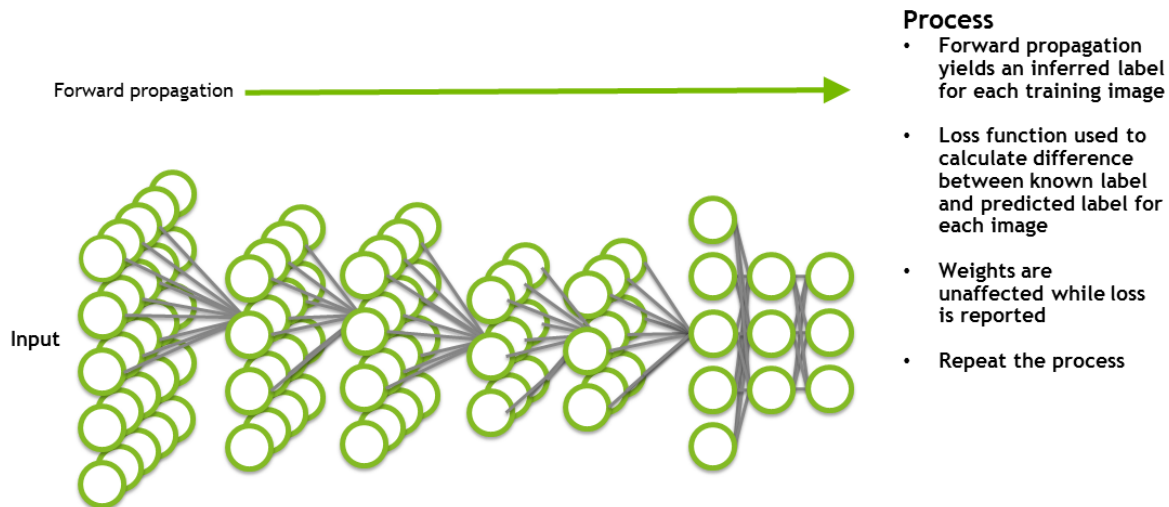
DEEP LEARNING APPROACH - TRAINING



성능 측정을 위한 검증 데이터 사용

검증 데이터 집합을 이용하면 인간 학습이 할 수 없는 기술을 사용하여 새로운 데이터에 대한 성능을 평가할 수 있습니다. 검증 데이터는 출력을 생성하기 위해 신경망에 입력되는데, 신경망은 이 데이터부터 *어떠한 학습도 하지 않습니다*. 손실은 계산되지만 모델 자체는 변하지 않습니다! 동일한 검증 데이터 세트를 마치 신규 검증 데이터처럼 취급하면서, 신규 데이터에 대한 성능을 반복해서 평가할 수 있습니다.

DEEP LEARNING APPROACH - VALIDATION



데이터 세트를 한 번 불러들였으면, 생성된 히스토그램에 마우스를 갖다 대거나 "DB 살펴보기(Exploring the db)"를 선택하여 실제 데이터를 살펴 보면서 자유로이 탐구하세요.

이제 여러분은 DIGITS에 데이터를 불러들여오는 방법을 배웠습니다. 여러분이 어디에서 신경망을 학습시키건 동일한 절차를 밟아야 합니다.

학습

다음 과제는 루이 데이터 세트를 가지고 했던 것과 같은 방식으로 신경망을 학습시키는 것입니다.

좌상단의 "DIGITS"를 선택하여 DIGITS 홈 화면에서부터 시작하세요. 다음으로는 새 모델(New Model) 과 분류(Classification) 를 선택해서 새로운 분류 모델을 만드세요. 여기부터는 여러분이 아래 정보를 가지고 적은 수의 데이터 세트를 훈련시켰던 것과 동일한 절차를 밟습니다.

"AlexNet" 신경망을 여러분이 불러들인 데이터로 5 에포크 동안 학습시킵니다.

여러분이 위 작업을 혼자서 잘 수행했다면 아래 작업을 잘 할 수 있다는 뜻입니다.

이미지 분류 신경망을 이미 불러들인 데이터로 학습시킵니다.

지난번 과제를 한 번 보실 필요가 있다면, [Train a Model. \(.../task1/task/Train%20a%20Model.ipynb\)](#)를 읽어 보세요.

이제 신경망을 학습시키는 중에 생성된 그래프를 살펴 보세요. 결과를 해석해 보세요. 이에 관해 곧 논의할 것입니다. 핵심 질문은 이것입니다. **모델이 학습을 진행하는 중에 성능을 평가하기 위해 그래프를 어떻게 사용해야 할까?**

주의: 학습 시간이 15 분 이상 걸린다면 아마도 5 에포크 이상으로 지정되어 있을지 모릅니다.

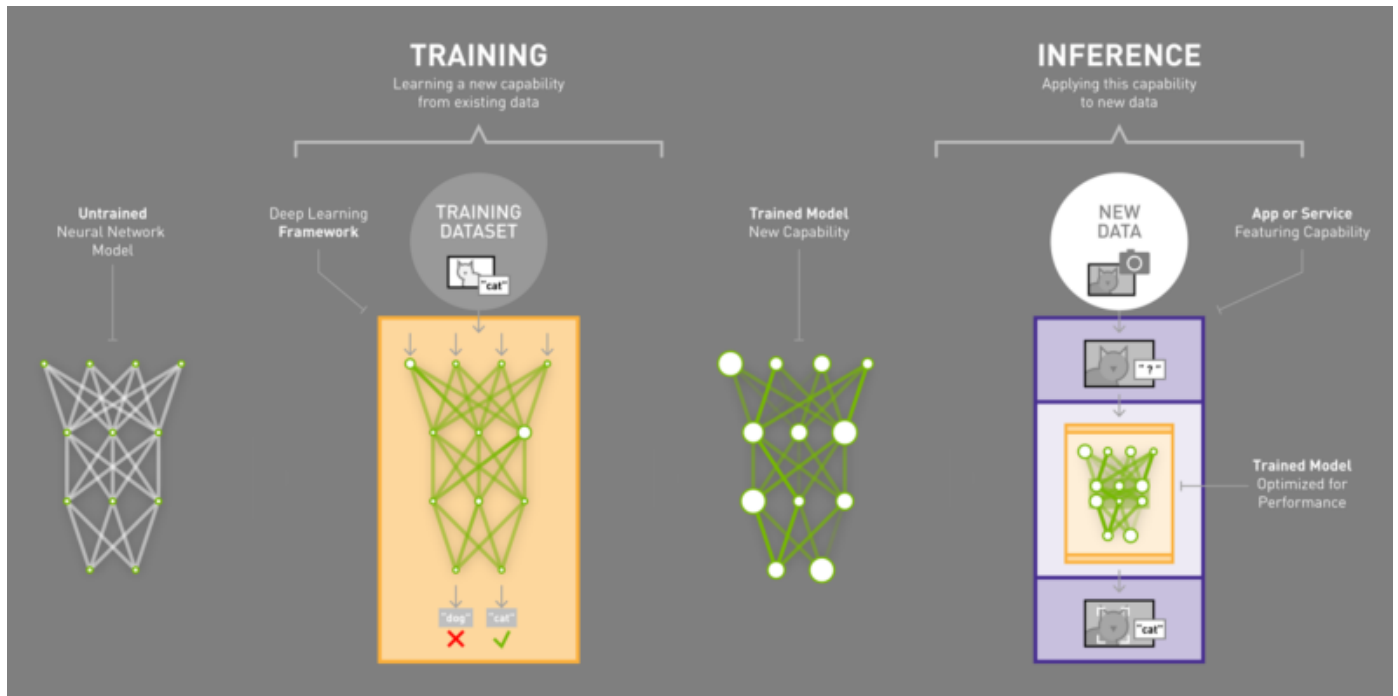
화면 상단에서 '작업 취소(Abort Job)'를 하고 '작업 복제(Clone Job)'를 한 후, 위 지시사항을 제대로 따랐는지 점검하세요.

학습이 진행되는 중에 아래 섹션을 읽으세요.

추론

여러분 모델의 작업 페이지를 보세요. 주 강좌로 돌아가서 이 페이지의 내용을 어떻게 해석하는지 자세히 살펴볼 것입니다. 지금은 우선 신경망이 이전에는 보지 못했던 이미지를 테스트하여 성공했는지 확인해 봅시다.

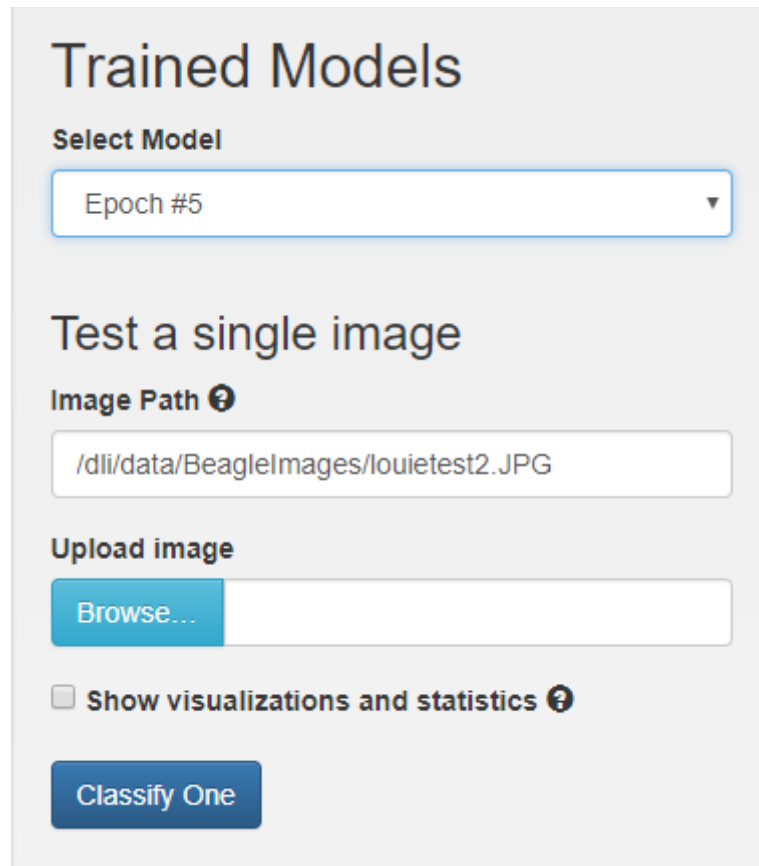
추론(inference)이란 학습에 근거하여 결정을 내리는 과정을 말합니다. 학습된 모델의 힘은 **레이블이 없는** 이미지를 분류하는 능력에서 나옵니다.



모델 윈도우 아래에서 단일 이미지나 이미지 목록을 테스트할 수 있습니다. 왼쪽의 이미지 경로 텍스트 박스 아래 경로명을 입력하세요.

```
/dli/data/BeagleImages/louietest2.JPG
```

그리고 **Classify One** 버튼을 클릭하세요. 몇 초 후, 새 창이 뜨고 이미지와 분류에 대한 정보가 표시될 것입니다.



결과를 두 가지로 이해할 수 있습니다.

1) 동작합니다! 학습되지 않은 신경망을 가져다가 수천 개의 레이블 있는 이미지에 노출했더니, 통계적으로 유의미한 신뢰성을 가진, 개를 개라고 분류하는 출력을 얻었습니다. 축하합니다!

2) 아직 다 된 것은 아닙니다. 인간 학습자는 100% 신뢰성으로 개 이미지를 분류할 것입니다. 우리 모델은 아직 상당한 손실을 가지고 있습니다.

두 가지 모두 맞습니다. 하지만 지금만은 1 번에 대해 축하합시다. 여러분이 이 노트북을 닫자마자 우리는 바로 성능에 대해 들여다볼 것입니다.

다른 이미지 테스트하기

학습 세트에 포함되지 않은 우리의 원래 데이터 세트를 가지고 자유롭게 테스트해 보세요.

아래 셀에서 Shift+Enter를 눌러 명령을 실행하여 이미지의 파일 경로를 확인하세요. "Image Path"에 전체 경로를 입력하면 테스트할 수 있습니다.

In []:

```
!ls /dli/data/BeagleImages/Louie
```

어느 이미지 분류 작업에나 같은 작업 흐름이 적용된다는 것을 기억하세요. AlexNet을 훈련시켜 여러분의 사진을 제 사진과 구별할 수 있고, 필기체 숫자 이미지들을 서로 분류할 수 있고, 건강한 환자와 병든 환자를 구별할 수 있습니다. 다음 과제에서 제일 처음으로 이것들을 다룰 것입니다.

첫 번째 소개 과제를 잘 끝내셨지만, 아직도 배울 것이 많습니다.

강좌로 돌아가서 우리가 방금 한 것들의 의미를 살펴보겠습니다. 이 창을 닫으면 본 GPU 인스턴스를 "중지"하는 옵션을 볼 것입니다. 이 창을 닫고 해당 옵션을 선택하세요.



DEEP
LEARNING
INSTITUTE

(<https://www.nvidia.com/en-us/deep-learning-ai/education/>).