**Mixup**

[[1710.09412v2] mixup: Beyond Empirical Risk Minimization (arxiv.org)](https://arxiv.org/abs/1710.09412v2)

인공지능 최신논문을 보고 정리

딥러닝 모델 성능 향상을 위해, 모델의 아키텍쳐를 변화시키지 않고, training dataset의 이미지를 합성시킨 후 실제 training dataset으로 학습시킵니다. Mixup 알고리즘으로 data augmentation로 얻는 이점을 기대할 수 있습니다.

지도학습에 사용되는 레이블링 된 이미지 두 장을 합성하여 새로운 이미지를 생성합니다. 이와 같은 방식으로 data augmentation을 할 경우 이미지의 각 클래스 간 decision boundary가 부드럽게 되어서, 오버피팅을 줄일 수 있습니다.

|  |
| --- |
| ● 라벨 1  ● 라벨 1 예측  ● 라벨 0  ○ 라벨 0 예측 |

왼쪽은 기존 ERM(Empirical risk Minimization)을 사용하여 학습시킨 경우입니다.

오른쪽은 Mixup을 적용한 training dataset을 사용하여 학습시킨 경우입니다.

왼쪽은 한쪽 클래스로 분류하지만, 오른쪽은 라벨의 경계가 흐려져 더 유연하게 추론합니다.

|  |
| --- |
|  |

와 는 각각 두 이미지의 값이고, 와 는 one-hot label encoding 된 값들이다. 은 새로 생성된 이미지 값이고 는 새로 생성된 라벨 값이다. λ는 임의의 변수 0부터 1 사이의 값으로 두 data의 혼합 비율을 나타낸다. (λ ∈ [0, 1])

|  |
| --- |
|  |
| def mixup(x1, x2, y1, y2, lambda\_=0.8):  x = lambda\_ \* x1 + (1-lambda\_) \* x2  y = lambda\_ \* y1 + (1-lambda\_) \* y2  return x, y  x, y = mixup(cat\_img, dog\_img, np.array([1, 0]), np.array([0, 1]))  plt.axis('off') plt.imshow(x.astype(int)), y plt.show() |

Mixup 알고리즘을 파이썬 코드로 구현한 예시입니다.

|  |
| --- |
|  |