

커리큘럼 학습의 우선순위 정책에 따른 이미지 분류 성능 비교

정보통신학회 춘계종합학술 대회

손건희, 윤성현



I. 커리큘럼 학습이란?

II. 제안한 방법

III. 실험 설계

IV. 실험 결과 및 분석

V. 결 론

I. 커리큘럼 학습이란?

- 아이디어: 인간의 학습 순서처럼 모델 훈련



fig1. 어린 아이 학습 모습

인간의 학습 순서처럼
모델 훈련



Easy-First



fig2. 어른 학습 모습

[fig1 출처] https://d14cvuwsb9oabg.cloudfront.net/c_fill,fl_lossy,w_960/v1475390407/ylgvnrjqwl1w5hzipjmxm.jpg

[fig2 출처]:https://mblogthumb-phinf.pstatic.net/MjAxODAzMjNfMjYy/MDAxNTIxNzcxMTQ1OTgx.YktnvvR5KY9lZlpluEHZ36HZCyU1zw0Mcs2tz65EMwog.q2kyKYiQrqVb3A4QXTJ-EVKkuTwwlclVye01zP8z-TAg.JPEG.dpdlddpf1/20180323_164537.jpg?type=w800

I. 커리큘럼 학습이란?

- 아이디어: 학습 샘플의 중요도를 다르게 설정

→ 쉬운 샘플에 높은 가중치 부여
샘플 난도 (difficulty): 손실 함수 값

사진 출처 : https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcRrY6NHU0PqZeGE3lvJGH28RVtk3sH_7QkHw6C8sllaEg&s



I. 커리큘럼 학습이란?

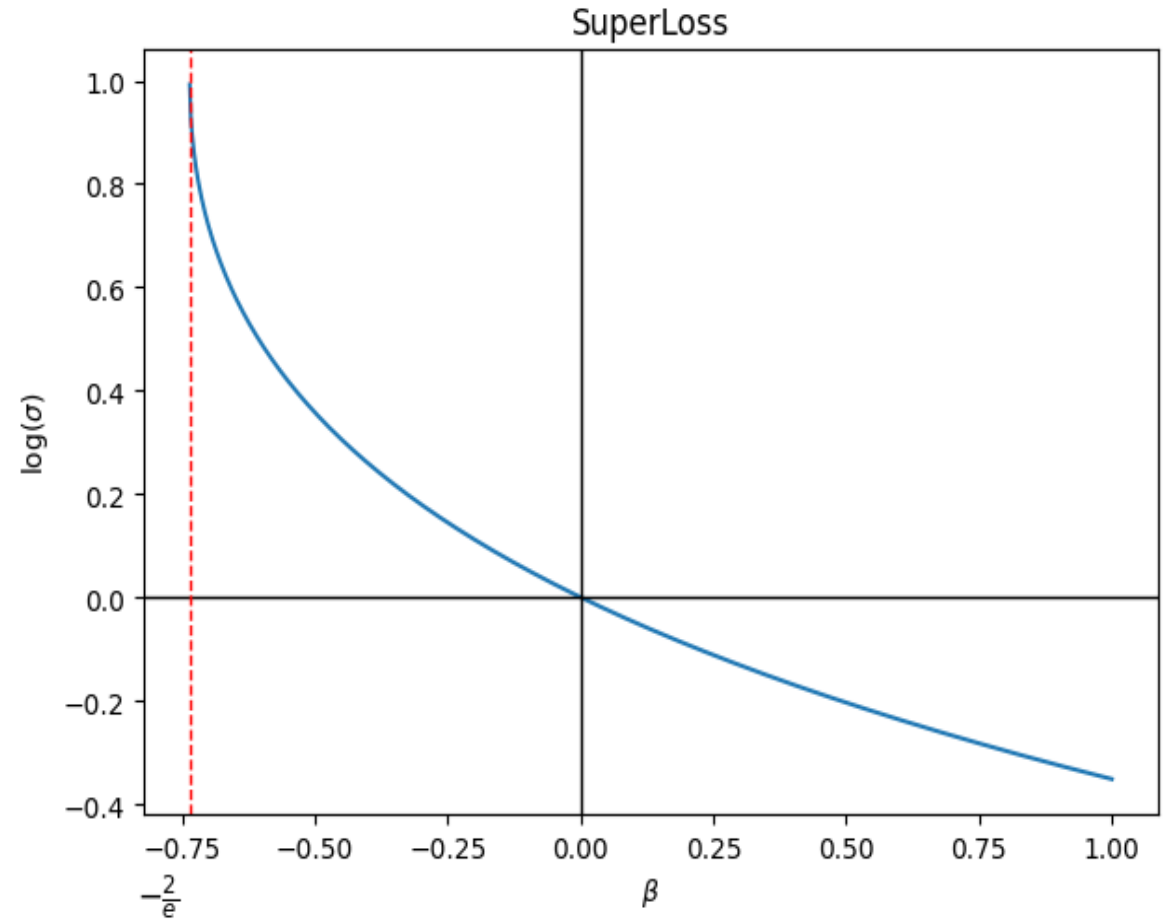
- Baseline : SuperLoss

Easy-First

$$SL(l_i) = (l_i - \tau)\sigma_i + \lambda(\log \sigma_i)^2$$

$$\sigma_i^*(l_i) = \operatorname{argmin}_{\sigma_i} SL(l_i) = e^{-W\left(\max\left(-\frac{1}{e}, \frac{\beta}{2}\right)\right)}$$

$$\beta = \frac{l_i - \tau}{\lambda}$$



I. 커리큘럼 학습이란?

- 기존의 SuperLoss의 한계점 : **Easy-First only**
- 다른 우선순위 정책들
 - **Hard-First**
 - **Medium-First**
 - ...

[Which Samples Should Be Learned First: Easy or Hard? | IEEE Journals & Magazine | IEEE Xplore](#)

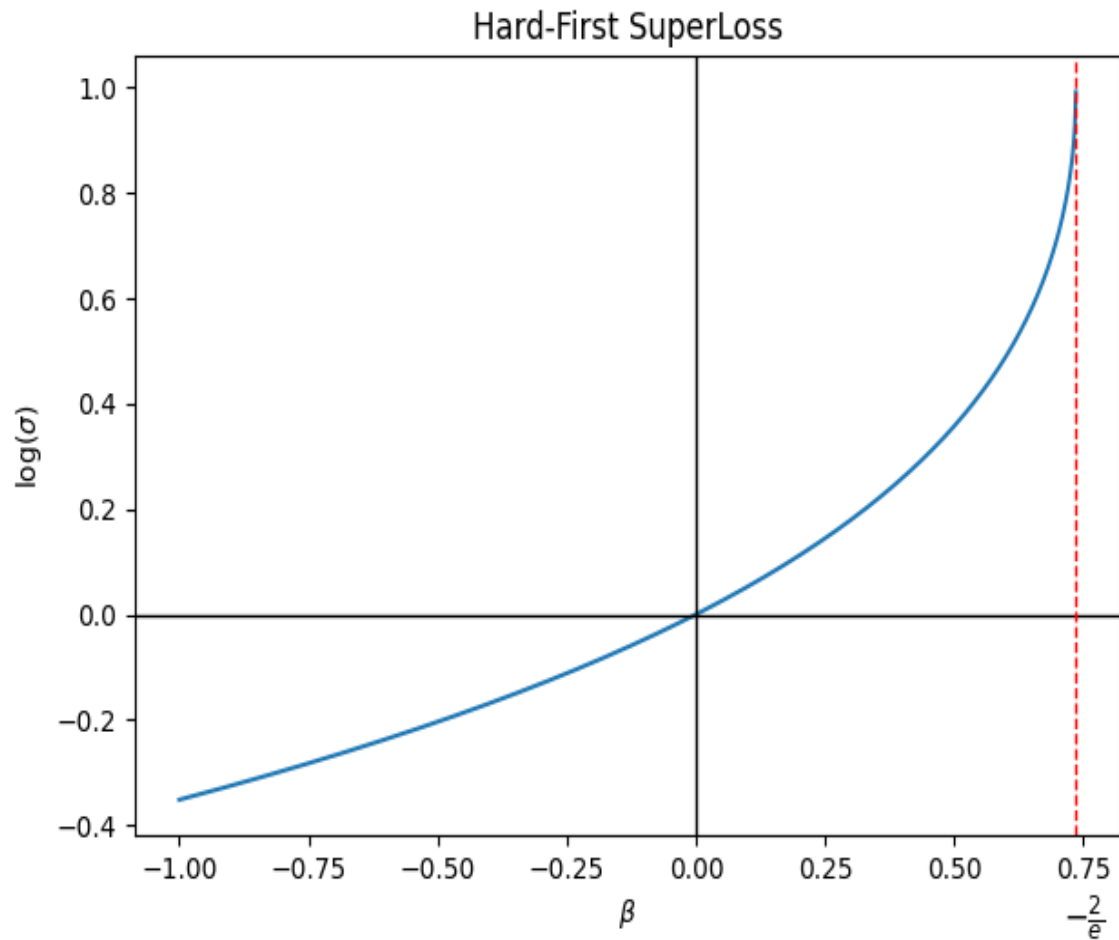
II. 제안한 방법

1. Hard-First

$$\text{HFSL}(l_i) = (l_i - \tau)\sigma_i - \lambda(\log \sigma_i)^2$$

$$\sigma_i^*(l_i) = \underset{\sigma_i}{\operatorname{argmin}} \text{HFSL}(l_i) = e^{-W\left(-\min\left(\frac{1}{e}, \frac{\beta}{2}\right)\right)}$$

$$\beta = \frac{l_i - \tau}{\lambda}$$



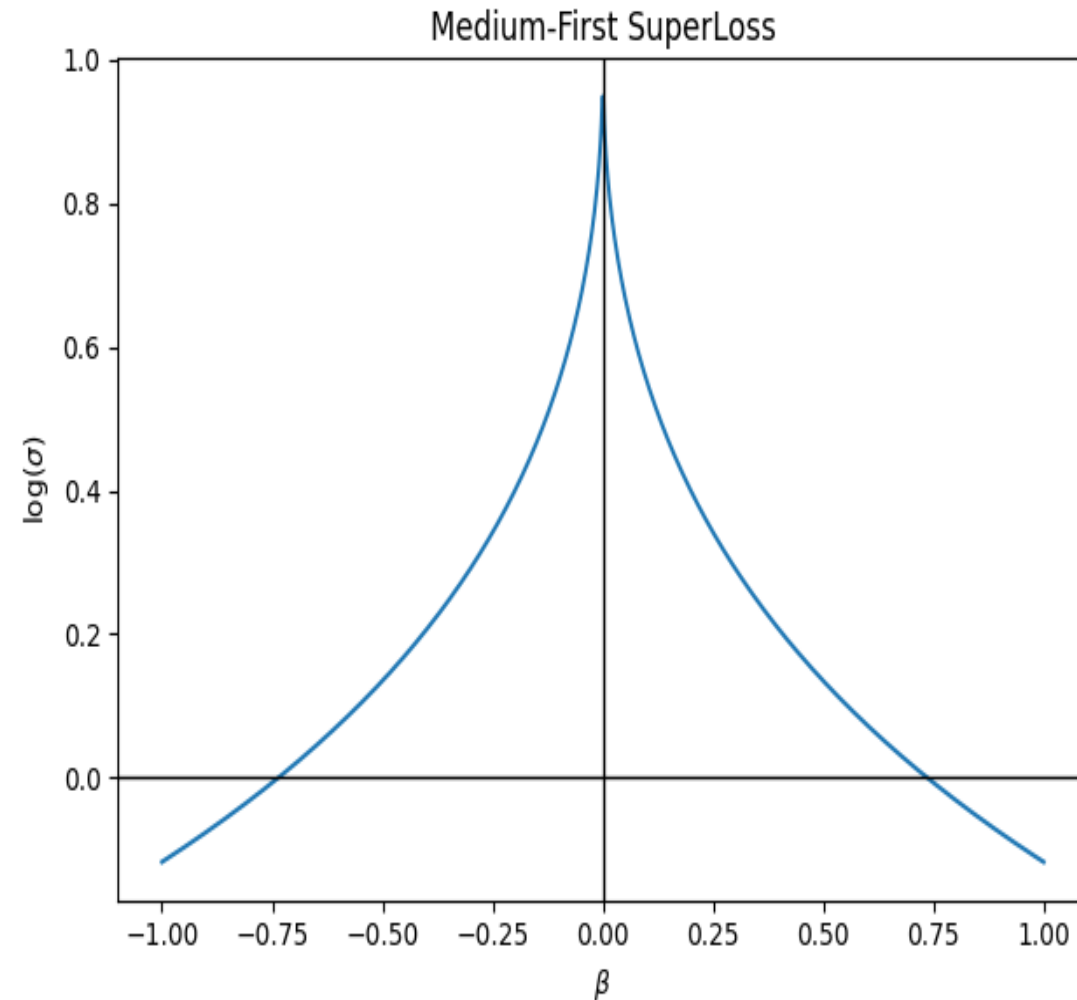
II. 제안한 방법

2. Medium-First

$$\text{MFSL}(l_i) = (l_i - \tau)\sigma_i + \text{sign}(l_i - \tau) \lambda \left\{ (\log \sigma_i)^2 - \frac{2\sigma}{e} \right\}$$

$$\begin{aligned} \sigma_i^*(l_i) &= \underset{\sigma_i}{\text{argmin}} \text{MFSL}(l_i) = e^{-W\left(\text{sign}(l_i - \tau) \left\{ \frac{\beta}{2} - \text{sign}(l_i - \tau) \frac{1}{e} \right\}\right)} \\ &= \begin{cases} e^{-W\left(-\frac{\beta}{2} - \frac{1}{e}\right)} & (l_i - \tau < 0) \\ e^{-W\left(\frac{\beta}{2} - \frac{1}{e}\right)} & (l_i - \tau \geq 0) \end{cases} \end{aligned}$$

$$\text{sign}(l_i - \tau) = \begin{cases} -1 & (l_i - \tau < 0) \\ 1 & (l_i - \tau \geq 0) \end{cases}$$



II. 제안한 방법

3. 우선순위 정책변경

예시 : Easy-to-Hard

$$MCSL_{EF \rightarrow HF}(l_i) = \begin{cases} SL(l_i) & (e_c \leq e_{thr}) \\ HFSL(l_i) & (e_c > e_{thr}) \end{cases}$$

Ⅲ. 실험 설계

실험 설계

Dataset : Cifar10

Base Model : CCT(compact convolutional transformer)

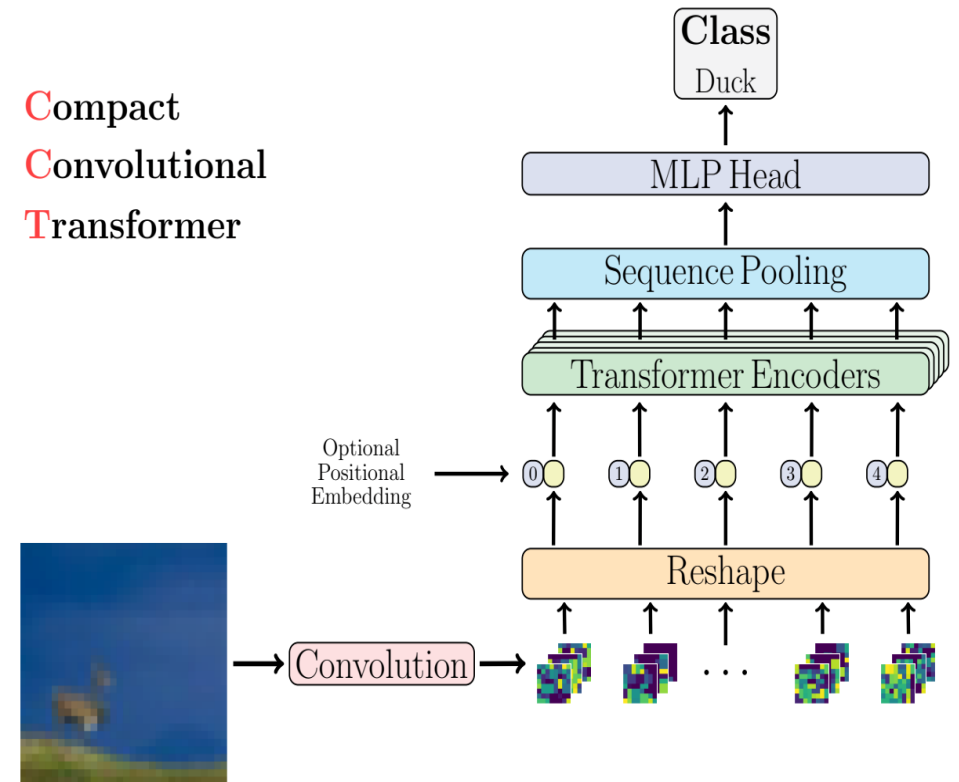
Label noise 추가 : 10%, 30%, 50%

학습률 : 0.0005

배치 크기 : 128

Epoch : 100

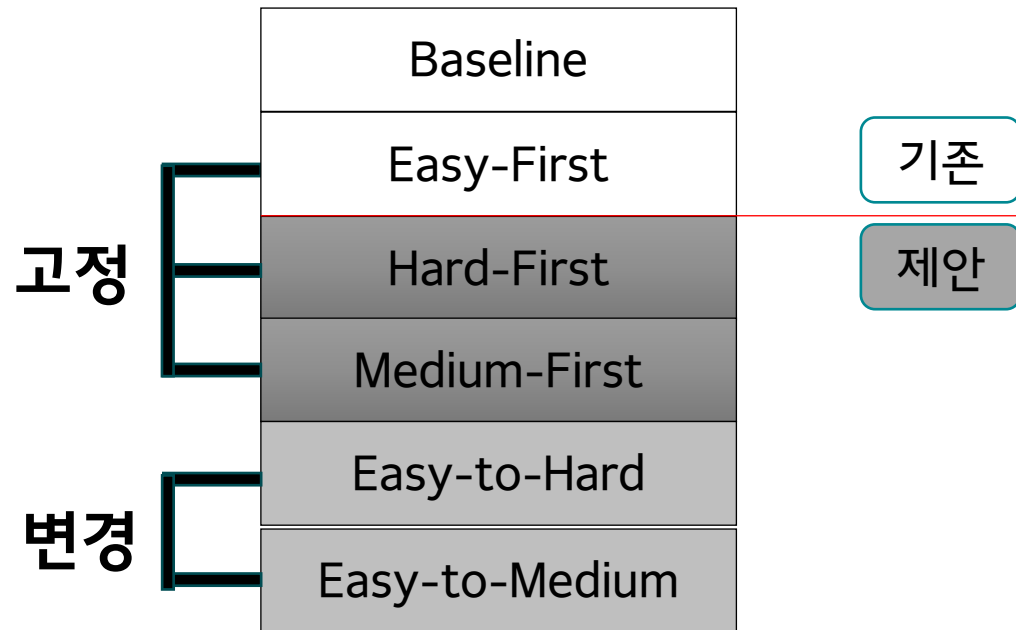
최적화 함수 : RAdam



https://production-media.paperswithcode.com/methods/cct_sBBD6Sv.png

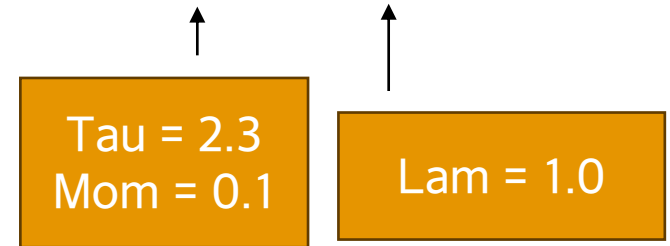
Ⅲ. 실험 설계

Method

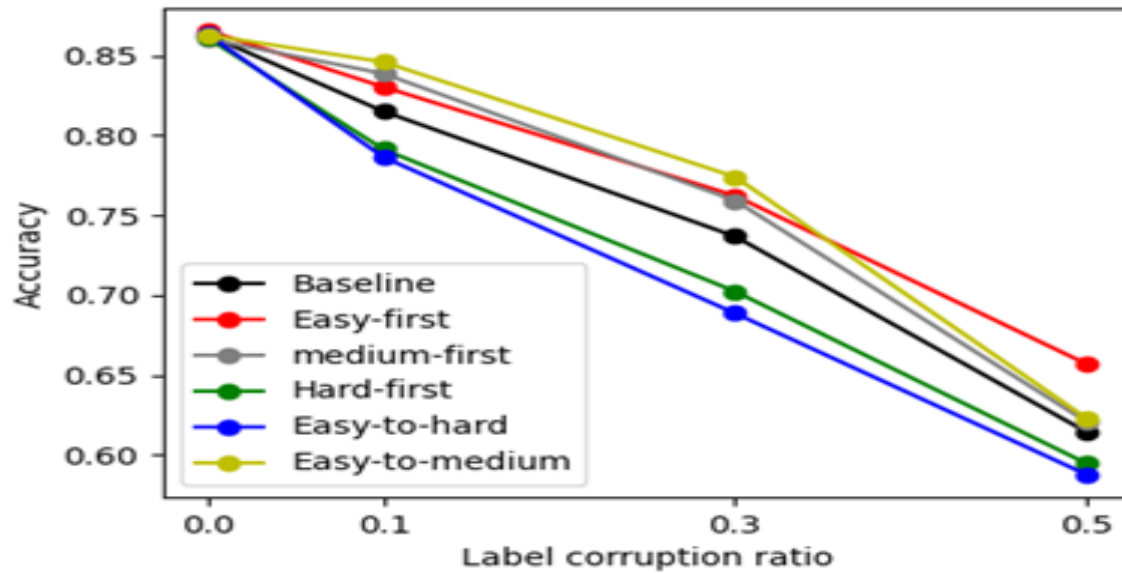


하이퍼 파라미터

$$SL(l_i) = (l_i - \tau)\sigma_i + \lambda(\log \sigma_i)^2$$



IV. 실험 결과 및 분석



Corruption \ Method	0%	10%	30%	50%
Baseline	0.86	0.81	0.74	0.61
Easy-First	0.87	0.83	0.76	0.66
Hard-First	0.86	0.79	0.70	0.59
Medium-First	0.86	0.84	0.76	0.62
Easy-to-Hard	0.86	0.79	0.69	0.59
Easy-to-Medium	0.86	0.85	0.77	0.62

Label corruption ratio 에 따른 각 방법의 CIFAR-10 평가 세트에서의 분류 정확도

IV. 실험 결과 및 분석

실험 결과 분석

	Easy group (to which Easy-First assigns higher weights)	Medium group (to which Medium-First assigns higher weights)	Hard group (to which Hard-First assigns higher weights)
Without noise	Clean & easy	Clean & medium	Clean & hard
Weak noise (e.g., 10%,30%)	Clean & easy + clean & medium	Clean & hard + clean & medium	Noisy & all
Strong noise (e.g., 50%)	Clean & easy + clean & medium	Clean & hard + noisy & easy	Noisy & medium + Noisy & hard

노이즈 X

모든 방법의 성능이 유사함

노이즈 O

노이즈 10%, 30% :
Easy-to-Medium

노이즈 50% : Easy-First



노이즈 데이터에 성능 우수

감사합니다

Gunhee Son: songunhee5426@gmail.com
Sunghyun Yoon : syoon@kongju.ac.kr