2023 하계 세미나

Pseudo-labeling for SF-UDA and SSL

Sogang University
Vision & Display Systems Lab, Dept. of Electronic Engineering

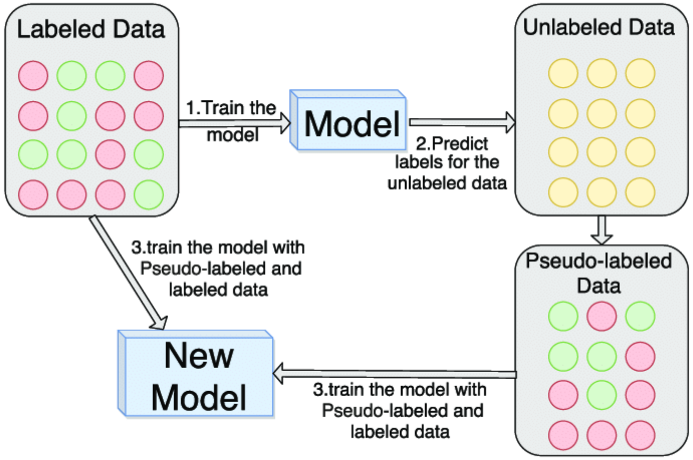
Presented By
박지원

Outline

- Background
  - Pseudo-labeling
- Guiding Pseudo-labels with Uncertainty Estimation for Source-free Unsupervised Domain Adaptation
  - CVPR 2023
- InPL: Pseudo-labeling the Inliers First for Imbalanced Semi-supervised Learning
  - ICLR 2023

Background

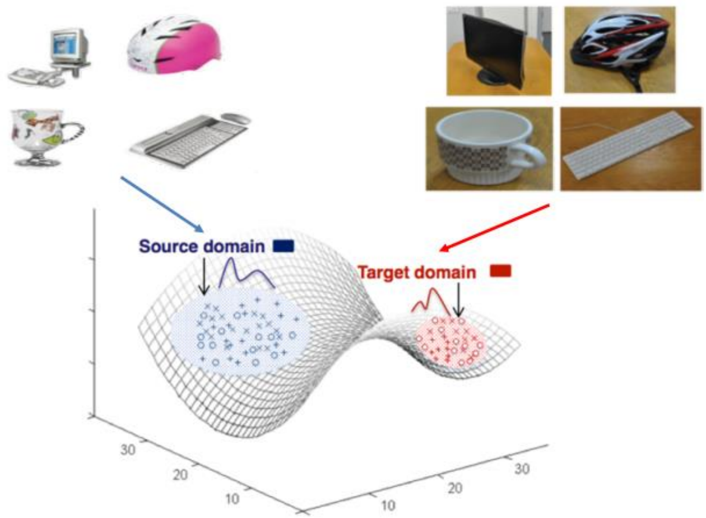
- Pseudo-labeling
  - 개념
    - Label이 없는 데이터에 대해 모델이 예측한 label을 사용하여 모델을 추가로 학습하는 방법
  - 학습 방식
    - Ground truth label이 있는 데이터로 모델 학습
    - 학습한 모델로 label 이 없는 데이터를 예측하고, 그 결과로 pseudo-label 생성
    - Pseudo-labeled data와 ground truth label data를 모두 사용하여 모델 학습



Litrico, Bue, et al. “Guiding Pseudo-labels with Uncertainty Estimation for Source-free Unsupervised Domain Adaptation.” CVPR, 2023.

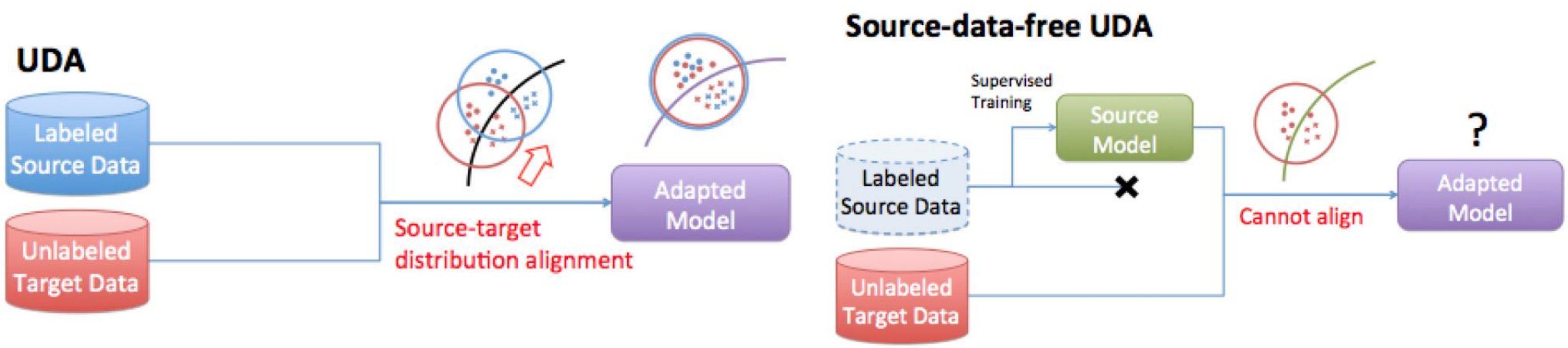
Background

- Domain Adaptation
  - 특정 domain 에서 학습된 모델을 다른 domain 으로 adapt 하려는 것
    - Source domain data: 모델이 학습하는 데이터
    - Target domain data: 평가 데이터
  - Domain gap: source domain 과 target domain 의 분포 상의 차이
  - 즉 Source domain과 target domain의 domain gap을 줄여 효율적인 학습을 진행하는 과정



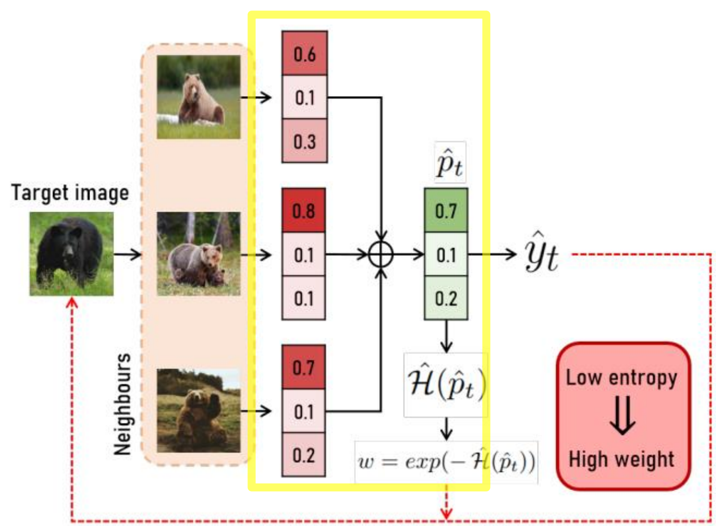
Background

- Source-free Unsupervised Domain Adaptation
  - Unsupervised Domain Adaptation(UDA)
    - 타겟 도메인의 데이터가 라벨 없이도 task 를 구행할 수 있도록 학습시킴
  - Source-free UDA
    - Source model 과 라벨이 없는 target data 를 통해 target domain 에 adapting 하는 방법론



1. Pseudo-label refinement

- Nearest neighbours knowledge aggregation
  - Target feature space 생성
    - Weakly augmented target samples 로부터 (features, predictions) pair 생성
    - Features 간의 cosine similarity 계산을 통해 neighbours 결정
  - Pseudo label refine
    - Neighbor 샘플의 prediction scores 를 평균내어 average score vector $\hat{p}_t$ 계산
    - Average score vector 의 max 값을 refined pseudo-label $\hat{y}_t$결정



$$\hat{p}_t^{(c)} = \frac{1}{K} \sum_{i \in \mathcal{I}} p_i'^{(c)},$$

$$\hat{y}_t = \arg\max_c \hat{p}_t^{(c)}.$$

서강대학교 SOGANG UNIVERSITY

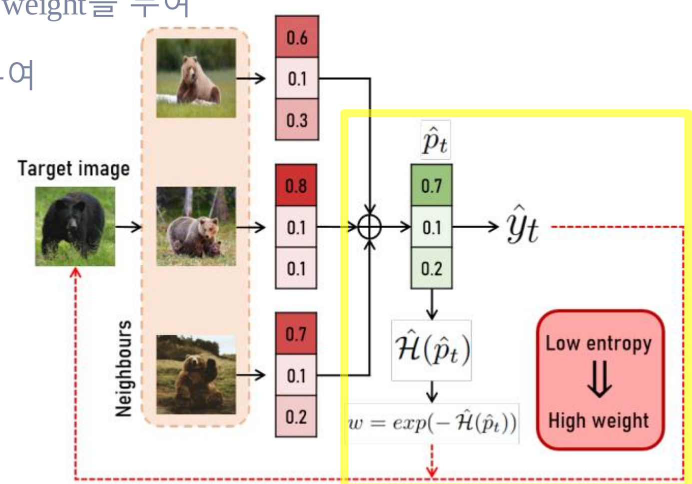
VDS LAB

2. Loss reweighting with uncertainty

- Entropy based uncertainty estimation
  - Loss weight define
    - Neighbour 샘플들에 대해 네트워크의 예측이 동일 → 해당 pseudo label 은 reliable (low uncertainty)
      - Classification loss term 에 high weight 적용
    - Neighbour 샘플들에 대해 네트워크의 예측이 서로 다름 → 해당 pseudo label 은 unreliable (high uncertainty)
      - Classification loss term 에 low weight 적용
  - Negative exponential function 사용
    - Exponential 함수의 입력으로 negative entropy 를 넣음
      - High entropy value 에 비해 low entropy value 에 더 큰 weight를 부여
      - Decision boundary 근처의 샘플에는 패널티를 적게 부여

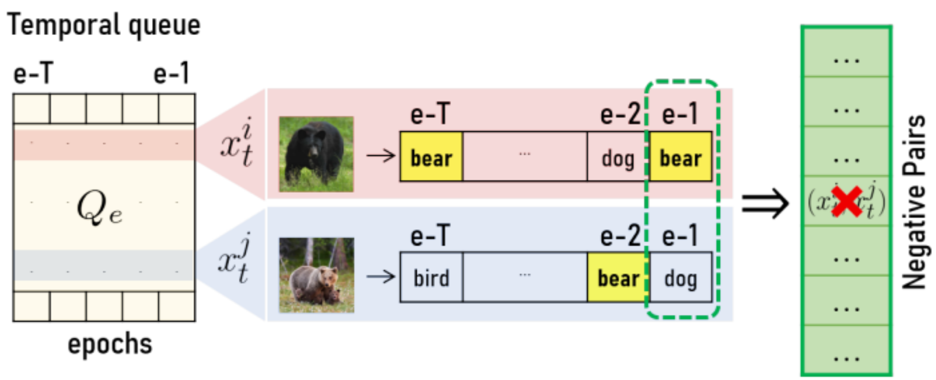
$$w_{x_t} = exp(-\hat{\mathcal{H}}(\hat{p}_t)).$$

$$L_t^{cls} = -\mathbb{E}_{x_t \in \mathcal{X}_t}\left[\boxed{w_{x_t}} \cdot \sum_{c=1}^{C} \tilde{y}^c \log\left(1 - p_{sa}^c\right)\right],$$



3. Temporal queue

- Self-supervised contrastive training for target data
  - Contrastive loss 에서 negative pairs 일부를 제외
    - 기존 SF-UDA methods 에서는 두 가지 augmentation sample pair 가 같은 pseudo label 을 공유하면 제외
      - Pseudo label의 noise 를 고려하지 못함
    - Pseudo-labels의 history 를 고려하는 방법 제시
      - T개의 past epochs 동안의 pseudo label를 queue Q_e 에 저장
      - Sample pair 가 T epoch 중에 한 번이라도 같은 pseudo-label 를 갖는다면 negative pairs 에서 제외



$$L_t^{ctr} = L_{\text{InfoNCE}} = -\log \frac{exp(q \cdot k_+/\tau)}{\sum_{j \in \boxed{\mathcal{N}_q}} exp(q \cdot k_j/\tau)}$$

서강대학교
SOGANG UNIVERSITY

VDS LAB

4. Self-learning with negative loss

- Joint training with self-learning
  - Negative learning loss 사용 – "NLNL: Negative Learning for Noisy Labels." ICCV 2019.
    - Pseudo-label refining 을 통해 pseudo-label의 정확도가 학습이 진행될수록 높아짐
    - 하지만 training 초기에는 pseudo-label 에 noise 가 존재
    - '입력 이미지가 어떤 레이블에 속하지 않는다' 라고 학습하는 방식
      - Pseudo-label 에 noise 가 많을 때 사용하는 학습 방식
      - 잘못된 레이블로 학습하는 것과 달리, 올바른 정보를 제공
      - 실험적으로 positive learning 보다 accuracy 가 높음을 확인

$$L_t^{cls} = -\,\mathbb{E}_{x_t \in \mathcal{X}_t}\left[w_{x_t} \cdot \sum_{c=1}^{C} \tilde{y}^c \log \boxed{\left(1 - p_{sa}^c\right)}\right]$$

Method	Acc.
Ours w/ positive	83.0
Ours w/ positive+negative	85.2
Ours	**90.0**

Figure 1: Conceptual comparison between *Positive Learning* (PL) and *Negative Learning* (NL). Regarding noisy data, while PL provides CNN the wrong information (red balloon), with a higher chance, NL can provide CNN the correct information (blue balloon) because a dog is clearly not a bird.

Experiments

- PACS dataset 실험 결과
  - Single target, multi-target 모두 높은 classification accuracy 보임

Single-Source UDA								
Method	SF-UDA	P → A	P → C	P → S	A → P	A → C	A → S	Avg.
NEL [1]	✓	82.6	80.5	32.3	98.4	84.3	56.1	72.4
Ours	✓	**87.5**	**84.2**	**75.8**	**98.8**	**84.6**	**77.2**	**84.7**

Table 1. Classification accuracy (%) on PACS for the single-source setting. All methods use the ResNet-18 backbone. Highest accuracies are in bold. We surpass the NEL [1] baseline by 12.3%.

Multi-Target UDA		P → ACS			A → PCS			
Method	SF-UDA	A	C	S	P	C	S	Avg.
1-NN	✗	15.2	18.1	25.6	22.7	19.7	22.7	20.7
ADDA [57]	✗	24.3	20.1	22.4	32.5	17.6	18.9	22.6
DSN [3]	✗	28.4	21.1	25.6	29.5	25.8	24.6	25.8
ITA [14]	✗	31.4	23.0	28.2	35.7	27.0	28.9	29.0
KD [44]	✗	24.6	32.2	33.8	35.6	46.6	57.5	46.6
NEL [1]	✓	**80.1**	**76.1**	25.9	96.0	**82.8**	49.8	68.4
Ours	✓	74.7	70.1	**68.7**	**94.6**	70.8	**71.5**	**75.0**

Table 2. Classification accuracy (%) on PACS for the multi-target setting. All methods use the ResNet-18 backbone. Highest accuracies are in bold. We surpass the SF-UDA baseline NEL [1] by 6.6%.

(a) Photo (b) Art painting (c) Cartoon (d) Sketch

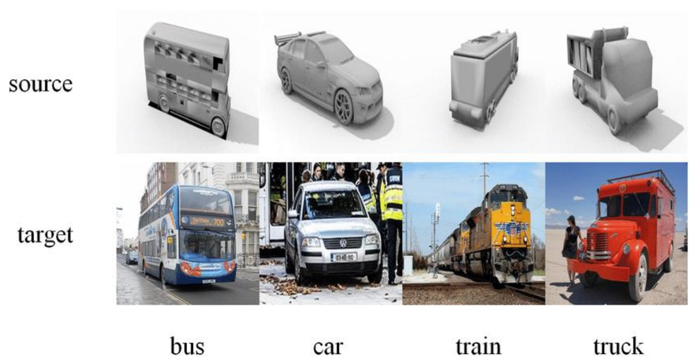
PACS 데이터셋 예시. 4개의 domain 과 7개의 object categories 로 구성

Experiments

- VisDA-C dataset 실험 결과

Method	SF-UDA	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Avg.
CDAN+BSP [9]	✗	92.4	61.0	81.0	57.5	89.0	80.6	90.1	77.0	84.2	77.9	82.1	38.4	75.9
SWD [30]	✗	90.8	82.5	81.7	70.5	91.7	69.5	86.3	77.5	87.4	63.6	85.6	29.2	76.4
MCC [25]	✗	88.7	80.3	80.5	71.5	90.1	93.2	85.0	71.6	89.4	73.8	85.0	36.9	78.8
CAN [26]	✗	**97.0**	87.2	82.5	74.3	**97.8**	96.2	90.8	80.7	96.6	**96.3**	87.5	59.9	87.2
DivideMix [35]	✓	95.0	82.4	85.3	78.1	94.2	90.3	90.1	81.3	92.5	91.9	91.2	60.8	86.1
SHOT [37]	✓	95.3	87.5	78.7	55.6	94.1	94.2	81.4	80.0	91.8	90.7	86.5	59.8	83.0
DIPE [62]	✓	95.2	87.6	78.8	55.9	93.9	95.0	84.1	81.7	92.1	88.9	85.4	58.0	83.1
NEL [1]	✓	94.5	60.8	**92.3**	87.3	87.3	93.2	87.6	**91.1**	56.9	83.4	93.7	**86.6**	84.2
A^2 Net [65]	✓	94.0	87.8	85.6	66.8	93.7	95.1	85.8	81.2	91.6	88.2	86.5	56.0	84.3
G-SFDA [66]	✓	96.1	88.3	85.5	74.1	97.1	95.4	89.5	79.4	95.4	92.9	89.1	42.6	85.4
SFDA-DE [11]	✓	95.3	**91.2**	77.5	72.1	95.7	**97.8**	85.5	86.1	95.5	93.0	86.3	61.6	86.5
AdaContrast [5]	✓	**97.0**	84.7	84.0	77.3	96.7	93.8	**91.9**	84.8	94.3	93.1	**94.1**	49.7	86.8
CoWA [32]	✓	96.8	90.3	87.0	67.4	97.2	96.6	90.4	87.3	95.6	95.5	91.8	62.5	88.2
Ours	✓	97.3	96.2	90.5	**91.8**	90.0	94.2	87.4	87.7	**97.0**	84.3	93.0	81.0	**90.0**

Table 3. Classification accuracy (%) on VisDA-C synthetic → real. All methods use the ResNet-101 backbone. The proposed approach outperforms the UDA state-of-the-art by 2.8% on average (Avg.) and the previous SF-UDA state-of-the-art by 1.8% on average (Avg.)



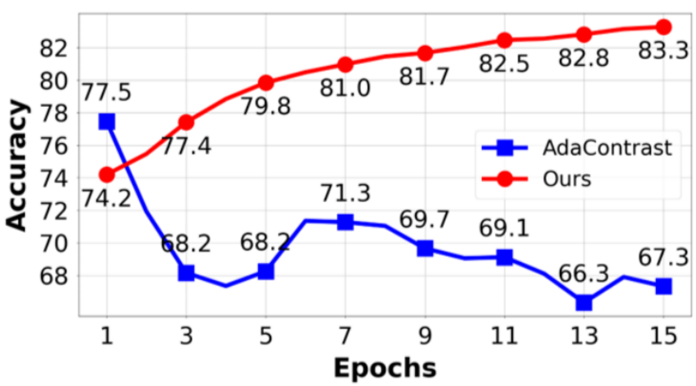
VisDA-C 데이터셋 예시.
Synthetic 이미지와 real 이미지를 포함하는
12개의 object categories 로 구성

Experiments

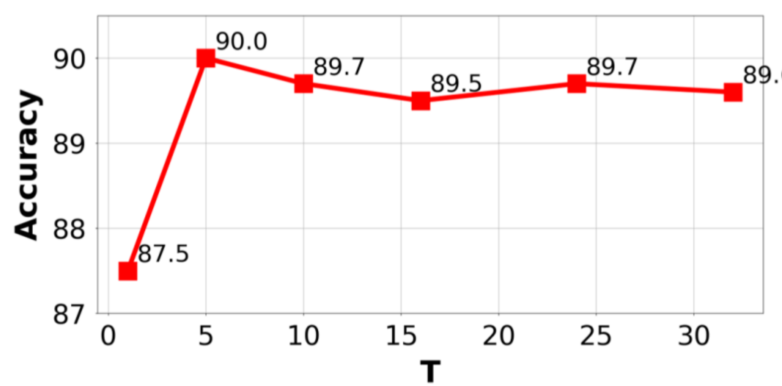
- Ablation Study

  - VisDA-C dataset 실험 결과

Pseudo-label refinement	Contrastive regularisation	Negative learning	Temporal-queue exclusion	Uncertainty reweighting	Avg. Acc.
✓	✗	✗	✗	✗	52.3
✓	✓	✗	✗	✗	78.9
✓	✓	✓	✗	✗	82.1
✓	✓	✓	✓	✗	85.8
✓	✓	✓	✓	✓	90.0



Refined pseudo-labels 의 classification accuracy
비교 모델인 AdaContrast 는 학습 초반의 pseudo label의 노이즈가 누적



Queue 의 length 에 따른 classification accuracy
T=5 에서 가장 높은 성능을 보임

서강대학교
SOGANG UNIVERSITY

VDS LAB

Yu, Li, et al. “InPL: Pseudo-labeling the Inliers First for Imbalanced Semi-supervised Learning.” ICLR, 2023.

Background

- Semi-supervised learning (SSL)
  - 소량의 labeled data 와 대용량의 unlabeled data 를 학습하는 방식
    - Labeled data 에는 supervised learning 적용, unlabeled data 에는 unsupervised learning 적용
  - Imbalaced SSL
    - 각 class의 data 수가 균일하지 않는 semi-supervised learning
    - Balanced SSL 에 비해 real-world scenario 에 적합함
  - General SSL methods
    - Supervised: multi-class cross-entropy loss
    - Unsupervised: consistency regularization & pseudo-labeling
      - Data augmentation 을 통해 생성한 데이터가 원래의 데이터와 같은 prediction(pseudo-label) 갖도록 학습

Background

- Consistency regularization with confidence-based pseudo-labeling

  - Process

    - Unlabeled data point x 에 대해 weak augmentation (w) 적용 후 model prediction

$$p(\mathbf{y}|\omega(\mathbf{x})) = f(\omega(\mathbf{x_b}))$$

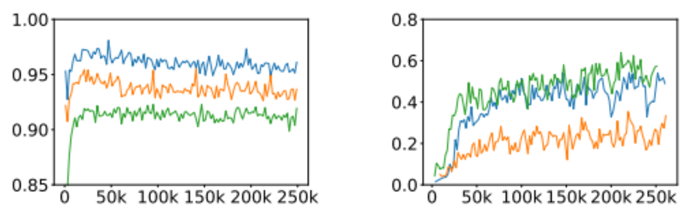
    - Maximum predicted 확률 $\max_i(p(y_i|w(x))$ 이 threshold τ_c 를 넘을 때에만 pseudo-label 생성
    - 생성한 pseudo-label 을 바탕으로 strong augmentation (Ω) 적용 데이터에 대해 모델 학습

$$\mathcal{L}_s = \frac{1}{B_s}\sum_{b=1}^{B_s}\mathcal{H}(\mathbf{y_b}, p(\mathbf{y}|\omega(\mathbf{x_b}))),$$ where $\mathcal{H}$ is the cross-entropy loss. ← supervised

$$\mathcal{L}_u = \frac{1}{B_u}\sum_{b=1}^{B_u}\mathbb{1}[\max_i(p(y_i|\omega(\mathbf{x_b}))) \geq \tau_c]\ \mathcal{H}(\hat{p}(\mathbf{y}|\omega(\mathbf{x_b})), p(\mathbf{y}|\Omega(\mathbf{x_b}))),$$ ← unsupervised

Background

- Confidence-based pseudo labeling 의 문제점
  - Confidence threshold 설정의 trade-off
    - High confidence threshold → minority classes 에 대한 pseudo-label 의 recall 이 낮음
    - Low confidence threshold → other classes 에 대한 pseudo-label 의 precision 이 낮음
  - Softmax-based confidence score 의 overconfident 문제
    - Out-of-distribution sample 에 대해 softmax-based confidence score 가 높은 경우가 있음
      - Low precision 으로 이어짐



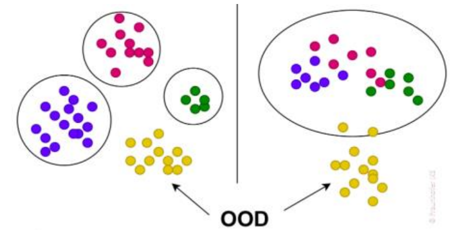
(a) Precision: Overall (d) Recall: Tail

Precision-Recall 분석 결과. (Tail: 가장 빈도가 낮은 3개의 classes)

파랑: InPL. 주황, 초록: softmax-based confidence score (FixMatch). 각각의 threshold 는 0.95, 0.6

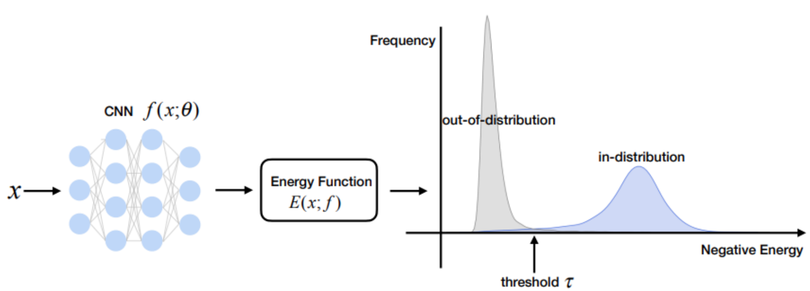
Inlier pseudo-labeling

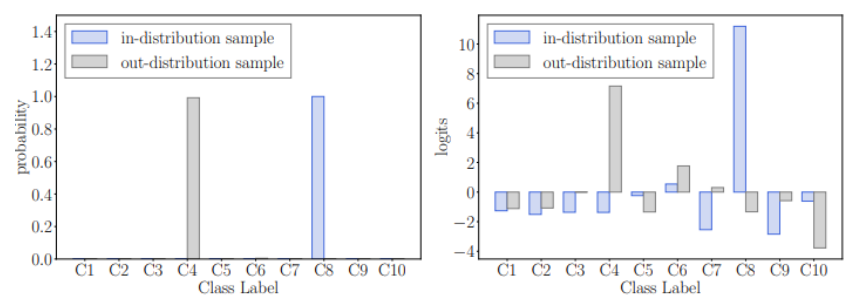
- Energy-based out-of-distribution detection
  - Train 과정
    - Known(observed) 데이터에 대해 낮은 에너지를 갖도록 train
    - Unknown(unobserved) 데이터에 대해서는 높은 에너지를 갖도록 train



OOD data의 분포 예시

  - Softmax 와 달리 energy score은 input의 probability density 와 일치
    - Overconfidence 문제에 덜 취약하여 OOD detect 성능 높음





(a) softmax scores 1.0 vs. 0.99 (b) negative energy scores: 11.19 vs. 7.11

Softmax score 과 energy score 비교

Inlier pseudo-labeling

- Energy score
  - Unlabeled sample 이 in-distribution 인지 out-of-distribution 인지 결정하기 위해 사용

$$E(\mathbf{x}, f(\mathbf{x})) = -T \cdot \log\left(\sum_{i=1}^{K} e^{f_i(\mathbf{x})/T}\right),$$

f : classifier

$f_i(x)$: i번째 class에 해당하는 logit value

T: temperature. (hyperparameter)

– Smaller energy scores → in-distribution

– Higher energy scores → out-of-distribution

  - Train 과정

– Unlabeled sample 에 대해 energy score 계산

– Threshold τ_e 보다 작은 energy score 갖는 sample 에 대해서만 pseudo-label 생성

$$\mathcal{L}_u = \frac{1}{B_u} \sum_{b=1}^{B_u} \mathbb{1}[E(\omega(\mathbf{x_b}), f(\omega(\mathbf{x_b}))) < \boxed{\tau_e}]\, \mathcal{H}(\hat{p}(\mathbf{y}|\omega(\mathbf{x_b})), p(\mathbf{y}|\Omega(\mathbf{x_b}))).$$

Inlier pseudo-labeling

- Energy score
  - Theoretical comparison confidence score vs. energy score
    - Negative log-likelihood loss 를 학습하는 과정에서 사용되는 gradient
    - In-distribution data point 의 energy score 가 작아지도록 학습되는 것을 알 수 있음

$$\frac{\partial \mathcal{L}_{\text{nll}}(x,y;\theta)}{\partial \theta} = \frac{1}{T}\frac{\partial E(x,y)}{\partial \theta} - \frac{1}{T}\sum_{j=1}^{K}\frac{\partial E(x,y)}{\partial \theta}\frac{e^{-E(x,y)/T}}{\sum_{j=1}^{K} e^{-E(x,j)/T}}$$

$$= \frac{1}{T}\Big(\underbrace{\frac{\partial E(x,y)}{\partial \theta}(1 - p(Y=y|x))}_{\downarrow \text{ energy pushed down for } y} - \underbrace{\sum_{j\neq y}\frac{\partial E(x,j)}{\partial \theta}p(Y=j|x)}_{\uparrow \text{ energy pulled up for other labels}}\Big).$$

x : in-distribution data

y : label

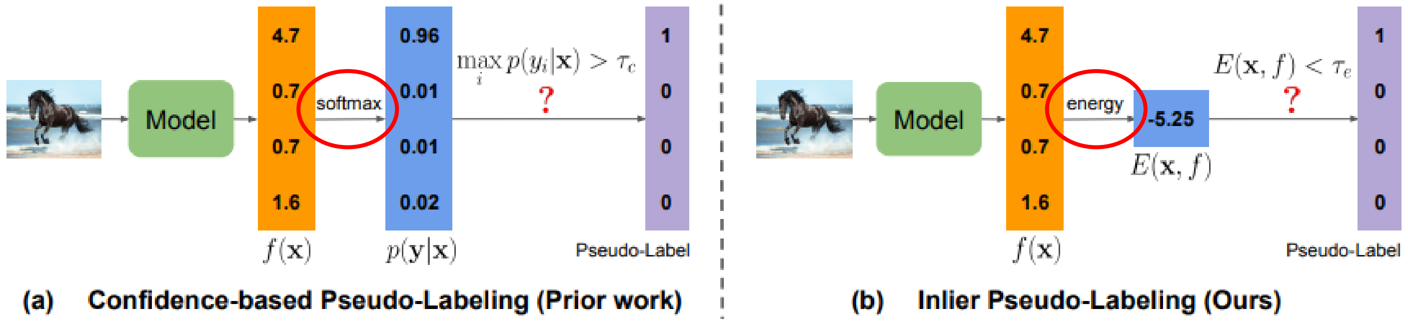
- 반면 log of max softmax confidence 를 살펴보면, energy score 가 상충됨

$$\log \max_y p(y \mid x) = E(x; f(x) - f^{\max}(x))$$

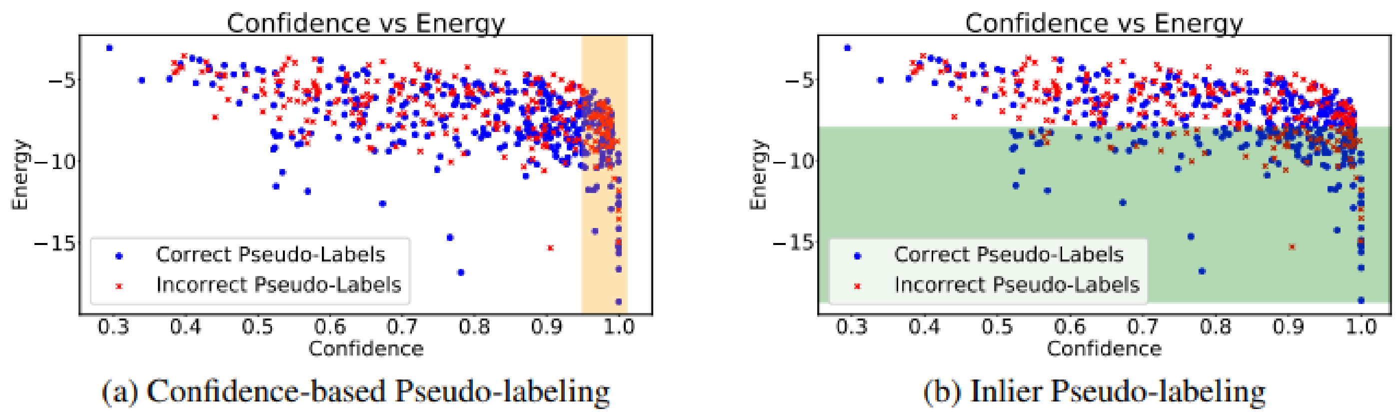
$$= \underbrace{E(x;f)}_{\downarrow \text{ for in-dist } x} + \underbrace{f^{\max}(x)}_{\uparrow \text{ for in-dist } x},$$

Inlier pseudo-labeling

- Energy score
  - Confidence-based pseudo-labeling 과의 비교



Overall framework 비교



Shaded region: unlabeled samples that are pseudo-labeled

Experiments

- Energy-based vs. confidence-based

	CIFAR10-LT			CIFAR100-LT	
	$\gamma = 50$	$\gamma = 100$	$\gamma = 200$	$\gamma = 50$	$\gamma = 100$
UDA (Xie et al., 2020a)	80.21±0.49	72.19±1.51	63.32±1.67	46.79±0.76	41.47±0.97
FixMatch (Sohn et al., 2020)	80.84±0.20	72.95±1.32	63.25±0.13	46.99±0.37	41.49±0.38
FixMatch-UPS (Rizve et al., 2021)	81.75±0.56	73.17±1.63	64.38±0.56	-	-
FixMatch-InPL w/o AML (ours)	**83.36**±0.38	**76.05**±0.84	**66.47**±1.06	**48.03**±0.31	**42.53**±0.68
FixMatch-Debias + AML (Wang et al., 2022)	83.53±0.67	76.92±1.72	67.70±0.44	**50.24**±0.46	44.12±0.81
FixMatch-InPL(ours)	**83.92**±0.52	**77.44**±1.17	**68.47**±1.15	49.96±0.36	**44.33**±0.61
OpenMatch (Saito et al., 2021)	81.01±0.45	73.15±1.03	63.22±1.86	46.92±0.28	40.76±0.81
FixMatch-D3SL (Guo et al., 2020)	81.20±0.33	72.71±2.32	65.09±1.72	46.83±0.45	41.22±0.39

Imbalanced SSL 실험 결과 (Top-1 accuracy)

(FixMatch-InPL: FixMatch framework에 InPL 적용)

Experiments

- Comparison to state-of-the-art imbalanced SSL methods

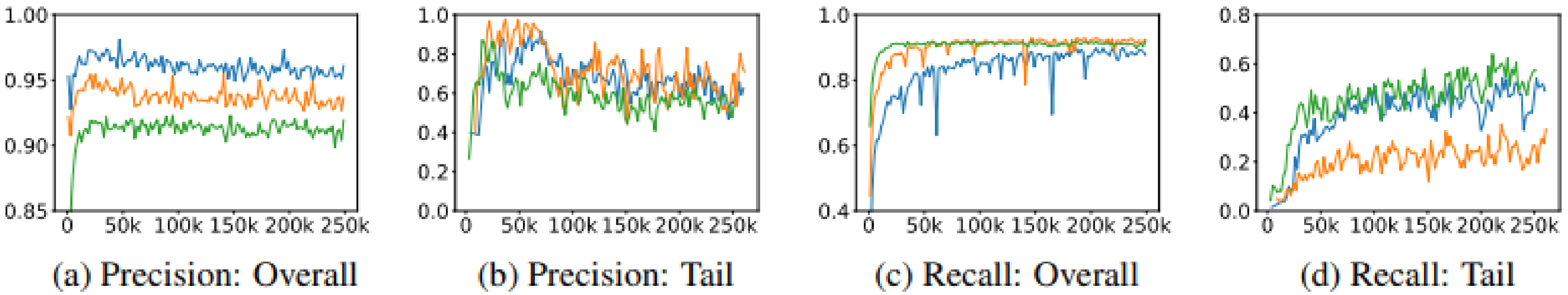
Dataset	CIFAR10-LT			CIFAR100-LT
Imbalance Ratio	$\gamma = 100$	$\gamma = 150$	$\gamma = 200$	$\gamma = 20$
FixMatch (Sohn et al., 2020)	72.3$\pm$0.33 / 53.8$\pm$0.63	68.5$\pm$0.60 / 45.8$\pm$1.15	66.3$\pm$0.49 / 42.4$\pm$0.94	51.0$\pm$0.20 / 32.8$\pm$0.41
w/ DARP+cRT (Kim et al., 2020)	78.1$\pm$0.89 / 66.6$\pm$1.55	73.2$\pm$0.85 / 57.1$\pm$1.13	-	54.7$\pm$0.46 / 41.2$\pm$0.42
w/ CReST+ (Wei et al., 2021)	76.6$\pm$0.46 / 61.4$\pm$0.85	70.0$\pm$0.82 / 49.4$\pm$1.52	-	51.6$\pm$0.29 / 36.4$\pm$0.46
w/ ABC (Lee et al., 2021)	81.1$\pm$0.82 / 72.0$\pm$1.77	77.1$\pm$0.46 / 64.4$\pm$0.92	73.9$\pm$1.18 / 58.1$\pm$2.72	56.3$\pm$0.19 / 43.4$\pm$0.42
w/ ABC-InPL (ours)	**82.9**$\pm$0.60 / **76.4**$\pm$1.49	**79.7**$\pm$0.71 / **70.8**$\pm$1.43	**76.4**$\pm$1.09 / **63.7**$\pm$2.03	**57.7**$\pm$0.33 / **46.4**$\pm$0.26
RemixMatch (Berthelot et al., 2020)	73.7$\pm$0.39 / 55.9$\pm$0.87	69.9$\pm$0.23 / 48.4$\pm$0.60	68.2$\pm$0.37 / 45.4$\pm$0.70	54.0$\pm$0.29 / 37.1$\pm$0.37
w/ DARP+cRT (Kim et al., 2020)	78.5$\pm$0.61 / 66.4$\pm$1.69	73.9$\pm$0.59 / 57.4$\pm$1.45	-	55.1$\pm$0.45 / 43.6$\pm$0.58
w/ CReST+ (Wei et al., 2021)	75.7$\pm$0.34 / 59.6$\pm$0.76	71.3$\pm$0.77 / 50.8$\pm$1.56	-	54.6$\pm$0.48 / 38.1$\pm$0.69
w/ ABC (Lee et al., 2021)	82.4$\pm$0.45 / 75.7$\pm$1.18	80.6$\pm$0.66 / 72.1$\pm$1.51	**78.8**$\pm$0.27 / 69.9$\pm$0.99	57.6$\pm$0.26 / 46.7$\pm$0.50
w/ ABC-InPL(ours)	**83.6**$\pm$0.45 / **81.7**$\pm$0.97	**81.3**$\pm$0.83 / **76.8**$\pm$0.88	**78.8**$\pm$0.75 / **74.5**$\pm$1.47	**58.4**$\pm$0.25 / **48.9**$\pm$0.36

Long-tailed dataset 실험 결과 (Top-1 accuracy)

(ABC-InPL: ABC(SoTA) framework에 InPL 적용)

Experiments

- Comparison to state-of-the-art imbalanced SSL methods
  - Precision을 크게 손상시키지 않으면서 tail class 에 대한 pseudo-label recall 을 약 2배 향상
    - InPL 이 tail class 에 대해 더 많은 true-positives 를 예측하고, head class 에 대해 less biased 되는 것
      - Head class: 가장 빈도가 높은 3개의 classes
      - Tail class: 가장 빈도가 낮은 3개의 classes



Precision-Recall 분석 결과.

파랑: InPL. 주황, 초록: softmax-based confidence score (FixMatch). 각각의 threshold 는 0.95, 0.6

감사합니다