CoMatch: Semi-supervised Learning with Contrastive Graph Regularization

Junnan Li Caiming Xiong Steven C.H. Hoi
Salesforce Research
{junnan.li,cxiong,shoi}@salesforce.com

ICCV-2021

Background

Semi-Supervised Learning

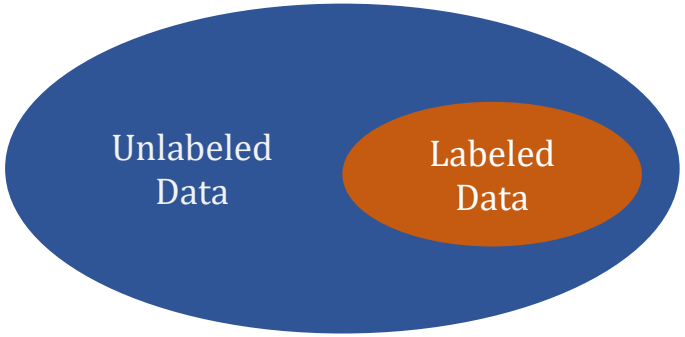
Semi-supervised learning has been an effective paradigm for leveraging unlabeled data to reduce the reliance on labeled data.

- **Consistency Regularization**

  The model should remains same or similar output distribution when add noise to input images. $\|p(y|\mathrm{Aug}(x)) - p(y|\mathrm{Aug}(x))\|_2^2$

- **Entropy Minimization**

  The entropy of the model on unlabeled data should be low as much as possible

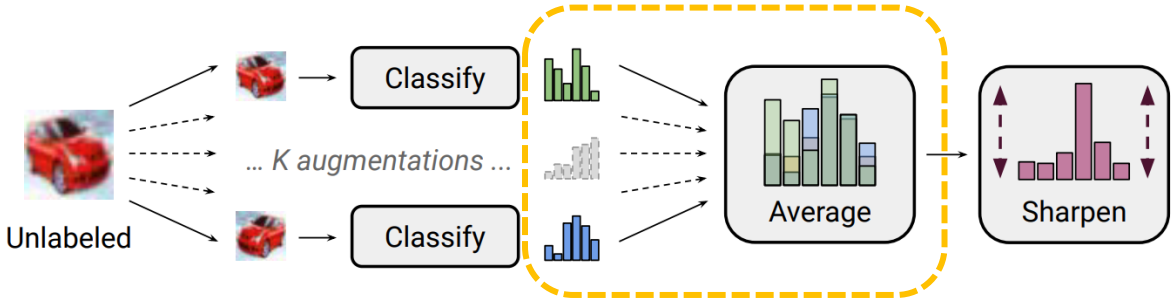


- **MixMatch** NeurIPS'19
- **ReMixMatch** ICLR'20
- **UDA** NeurIPS'20
- **FixMatch** NeurIPS'20

Background

- **MixMatch** NeurIPS'19

$$\bar{q}_b = \frac{1}{K}\sum_{k=1}^{K} \mathrm{p}_{\mathrm{model}}(y \mid \hat{u}_{b,k}; \theta)$$ **Artificial target**



Entropy minimization

$$\mathrm{Sharpen}(p, T)_i := p_i^{\frac{1}{T}} \Big/ \sum_{j=1}^{L} p_j^{\frac{1}{T}}$$

$\hat{\mathcal{X}} = ((\hat{x}_b, p_b); b \in (1, \ldots, B))$ *// Augmented labeled examples and their labels*

$\hat{\mathcal{U}} = ((\hat{u}_{b,k}, q_b); b \in (1, \ldots, B), k \in (1, \ldots, K))$ *// Augmented unlabeled examples, guessed labels*

$\mathcal{W} = \mathrm{Shuffle}(\mathrm{Concat}(\hat{\mathcal{X}}, \hat{\mathcal{U}}))$ *// Combine and shuffle labeled and unlabeled data*

$\mathcal{X}' = (\mathrm{MixUp}(\hat{\mathcal{X}}_i, \mathcal{W}_i); i \in (1, \ldots, |\hat{\mathcal{X}}|))$ *// Apply MixUp to labeled data and entries from $\mathcal{W}$*

$\mathcal{U}' = (\mathrm{MixUp}(\hat{\mathcal{U}}_i, \mathcal{W}_{i+|\hat{\mathcal{X}}|}); i \in (1, \ldots, |\hat{\mathcal{U}}|))$ *// Apply MixUp to unlabeled data and the rest of $\mathcal{W}$*

$$\mathcal{L}_{\mathcal{X}} = \frac{1}{|\mathcal{X}'|}\sum_{x,p \in \mathcal{X}'} \mathrm{H}(p, \mathrm{p}_{\mathrm{model}}(y \mid x; \theta))$$

$$\mathcal{L}_{\mathcal{U}} = \frac{1}{L|\mathcal{U}'|}\sum_{u,q \in \mathcal{U}'} \|q - \mathrm{p}_{\mathrm{model}}(y \mid u; \theta)\|_2^2$$

$$\mathcal{L} = \mathcal{L}_{\mathcal{X}} + \lambda_{\mathcal{U}}\mathcal{L}_{\mathcal{U}}$$

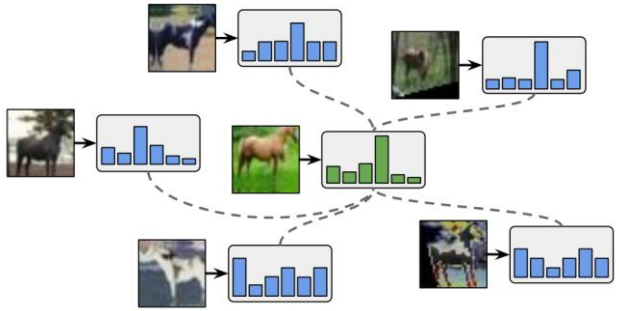
- **ReMixMatch** ICLR'20

a. Distribution Alignment

$$\tilde{q} = \mathrm{Normalize}\left(q \times \frac{p(y)}{\tilde{p}(y)}\right)$$

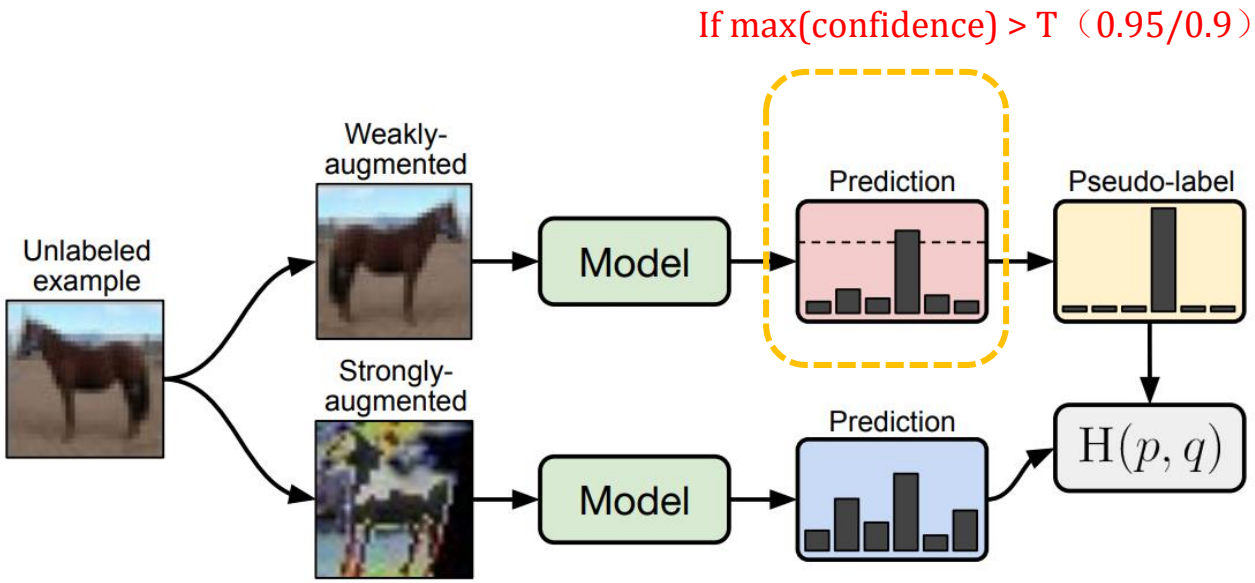
enforces that the aggregate of predictions on unlabeled data matches the distribution of the provided labeled data.

b. Augmentation Anchor



Background

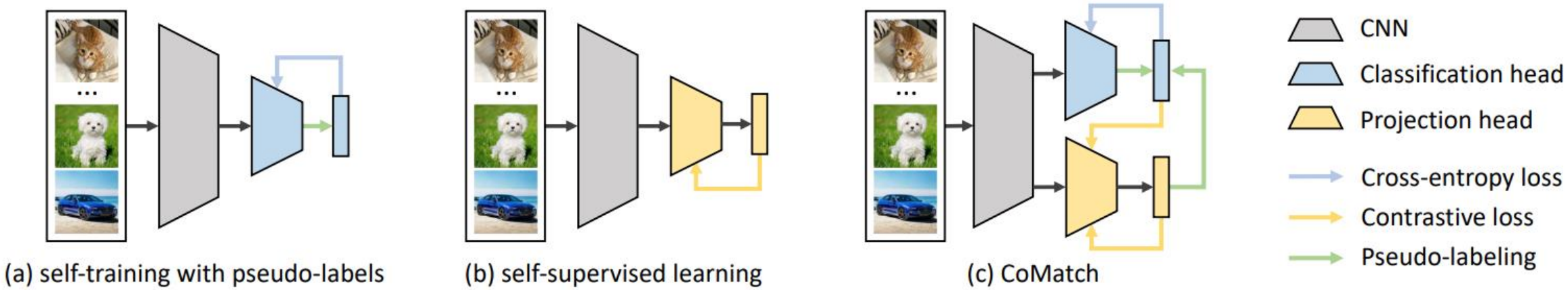
- **FixMatch** NeurIPS'20



$$\ell_s = \frac{1}{B}\sum_{b=1}^{B} \mathrm{H}(p_b, p_{\mathrm{m}}(y \mid \alpha(x_b)))$$

$$\ell_u = \frac{1}{\mu B}\sum_{b=1}^{\mu B} \mathbb{1}(\max(q_b) \geq \tau)\, \mathrm{H}(\hat{q}_b, p_{\mathrm{m}}(y \mid \mathcal{A}(u_b)))$$

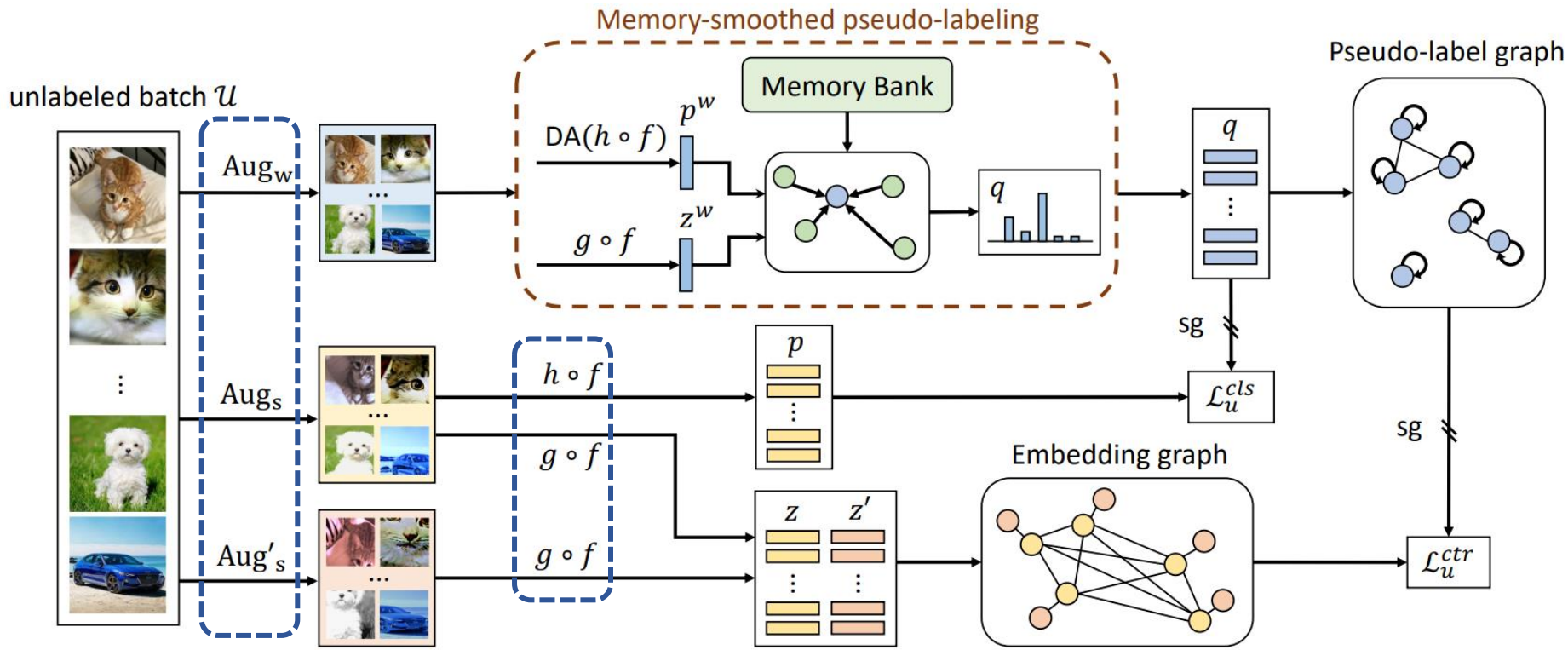
Motivation



- Pseudo-labeling (also called self-training) methods heavily rely on the quality of the model's class prediction, thus suffering from confirmation bias where the prediction mistakes would accumulate.
- Self-supervised learning (Pre-trained) methods are task-agnostic, and the widely adopted contrastive learning may learn representations that are suboptimal for the specific classification task.

CoMatch

consistency regularization + entropy minimization + contrastive learning + graph-based SSL

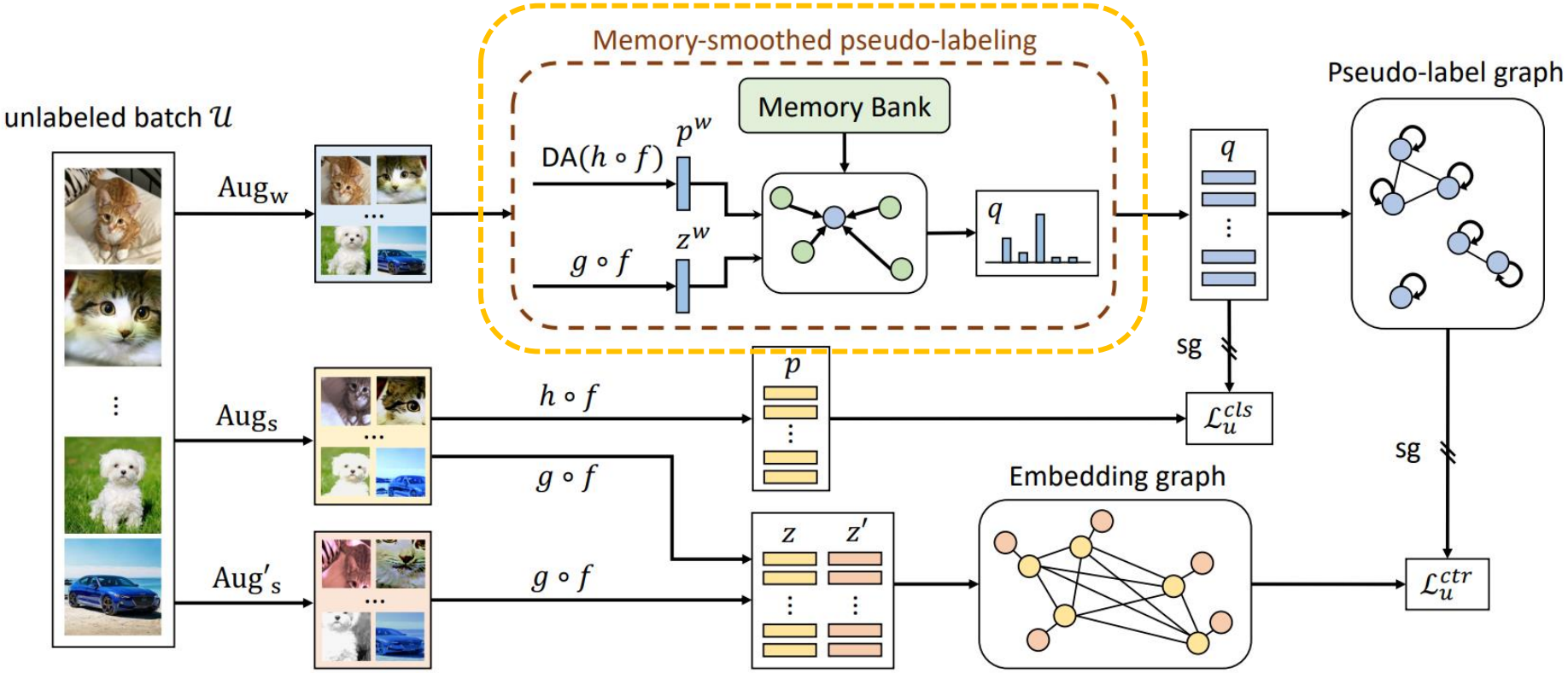


- CNN f
- Classification head h
- Projection head g

- Aug_w refers to weak augmentations
- Aug_s refers to strong augmentations

CoMatch

DA prevents the model's prediction from collapsing to certain classes.



- **Memory-smoothed pseudo-labeling**

moving-average $\tilde{p}^w$

$$p^w = h \circ f(\text{Aug}_\text{w}(u)) \longrightarrow p^w = \text{DA}(p^w) \longrightarrow p^w = \text{Normalize}(p^w / \tilde{p}^w)$$

The past **K** weakly-augmented samples $\longrightarrow \text{MB} = \{(p_k^w, z_k^w)\}_{k=1}^K$ **FIFO**

$$a_k = \frac{\exp(z_b^w \cdot z_k^w / t)}{\sum_{k=1}^K \exp(z_b^w \cdot z_k^w / t)}$$

Pseudo-labels

$$q_b = \alpha p_b^w + (1-\alpha) \sum_{k=1}^K a_k p_k^w$$

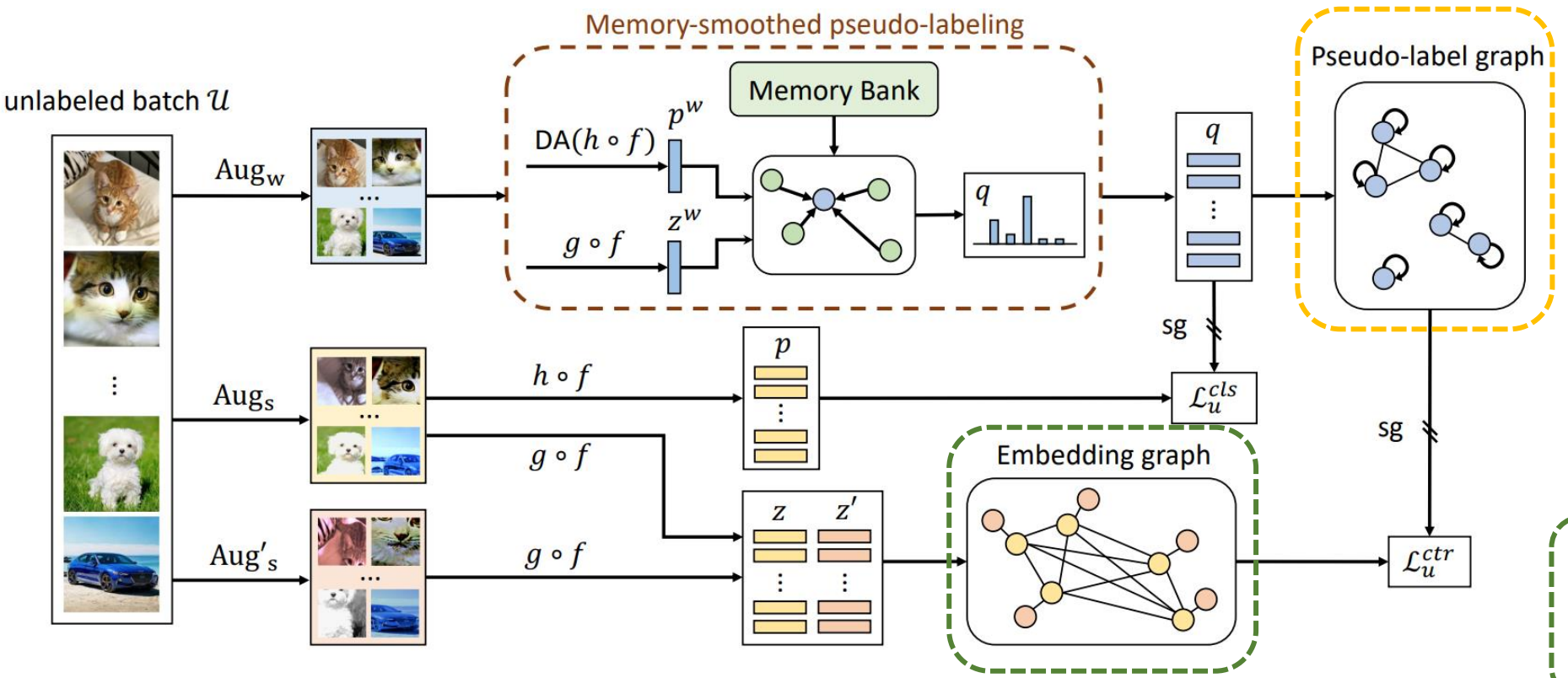
For unlabeled sample $\mathbf{u_b}$

$$J(q_b) = (1-\alpha) \sum_{k=1}^K a_k \| q_b - p_k^w \|_2^2 + \alpha \| q_b - p_b^w \|_2^2$$

CoMatch

- **Graph-based contrastive learning**

Size: $\mu B \times \mu B$



$$W_{bj}^q = \begin{cases} 1 & \text{if } b = j \\ q_b \cdot q_j & \text{if } b \neq j \text{ and } q_b \cdot q_j \geq T \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{W}_{bj} = W_{bj} / \textstyle\sum_j W_{bj}$$

$$\mathcal{L}_u^{ctr} = \frac{1}{\mu B} \sum_{b=1}^{\mu B} \mathrm{H}(\hat{W}_b^q, \hat{W}_b^z)$$

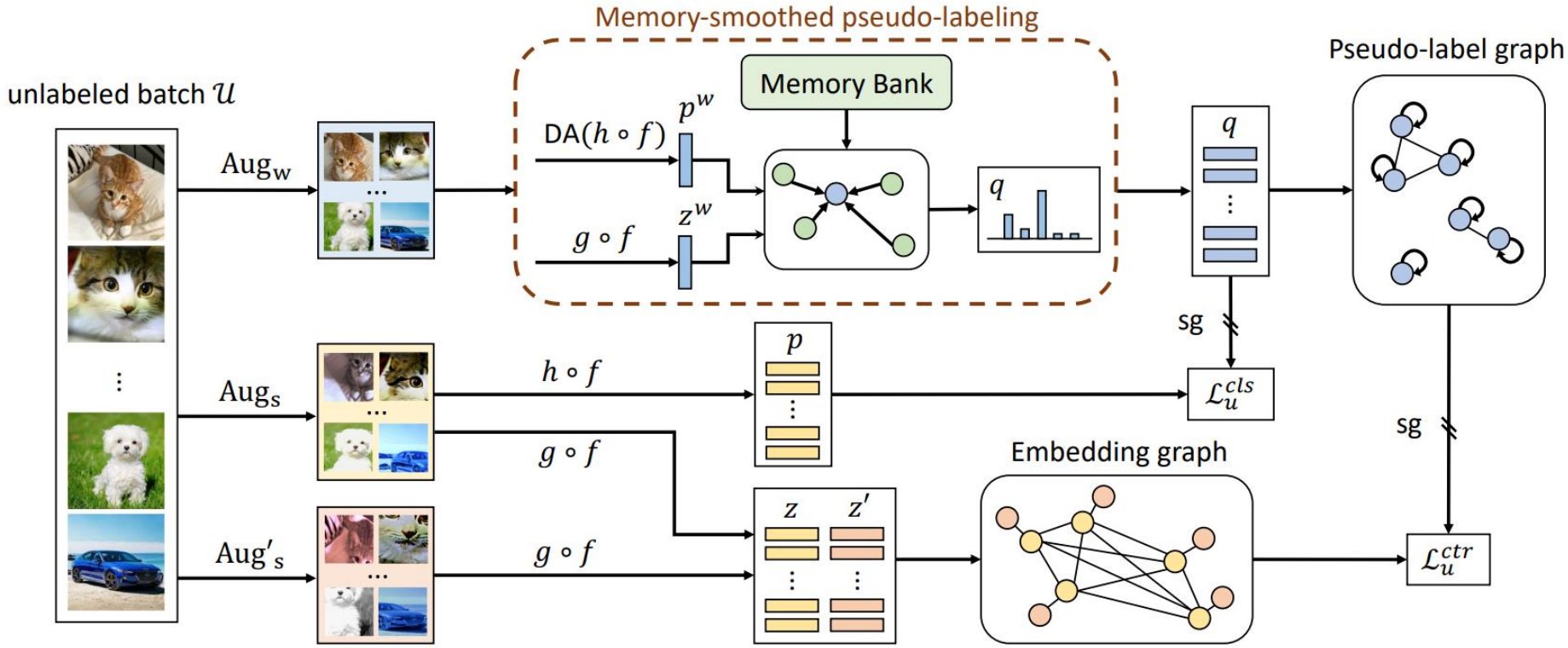
$$W_{bj}^z = \begin{cases} \exp(z_b \cdot z'_b / t) & \text{if } b = j \\ \exp(z_b \cdot z_j / t) & \text{if } b \neq j \end{cases}$$

$$\hat{W}_{bj} = W_{bj} / \textstyle\sum_j W_{bj}$$

$$-\hat{W}_{bb}^q \log\left(\frac{\exp(z_b \cdot z'_b / t)}{\sum_{j=1}^{\mu B} \hat{W}_{bj}^z}\right) - \sum_{j=1, j\neq b}^{\mu B} \hat{W}_{bj}^q \log\left(\frac{\exp(z_b \cdot z_j / t)}{\sum_{j=1}^{\mu B} \hat{W}_{bj}^z}\right)$$

consistency regularization — entropy minimization

CoMatch



- **Loss Function**

$$\mathcal{L}_x = \frac{1}{B}\sum_{b=1}^{B} \mathrm{H}(y_b, p(y|\mathrm{Aug_w}(x_b)))$$

$$\mathcal{L}_u^{cls} = \frac{1}{\mu B}\sum_{b=1}^{\mu B} \mathbb{1}(\max q_b \geq \tau)\mathrm{H}(q_b, p(y|\mathrm{Aug_s}(u_b)))$$

$$\mathcal{L}_u^{ctr} = \frac{1}{\mu B}\sum_{b=1}^{\mu B} \mathrm{H}(\hat{W}_b^q, \hat{W}_b^z)$$

Experiments

Method	CIFAR-10				STL-10
	20 labels	40 labels	80 labels	250 labels	1000 labels
MixMatch [2]	27.84±10.63	51.90±11.76	80.79±1.28	88.97±0.85	38.02±8.29
FixMatch [32]	82.32±9.77	86.12±3.53	92.06±0.88	94.90±0.67	65.38±0.42
FixMatch [32] w. DA [1]	83.81±9.35	86.98±3.40	92.29±0.86	94.95±0.66	66.53±0.39
CoMatch	**87.67**±8.47	**93.09**±1.39	**93.97**±0.62	**95.09**±0.33	**79.80**±0.38

Table 1: Accuracy for CIFAR-10 and STL-10 on 5 different folds. All methods are tested using the same data and codebase.

- For CIFAR-10: Wide ResNet-28-2
- For STL-10: ResNet-18
- Weak Augmentation: standard crop-and-flip
- Strong Augmentation: RandomAugment
- Strong Augmentation':

```
from torchvision import transforms as T
color_jitter = T.ColorJitter(0.4,0.4,0.4,0.1)
transforms.Compose([
  T.RandomApply([color_jitter], p=0.8)
  T.RandomGrayscale(p=0.2)])
```

Experiments

Self-supervised Pre-training	Method	#Epochs	#Paramters (train/test)	Top-1 Label fraction 1%	Top-1 Label fraction 10%	Top-5 Label fraction 1%	Top-5 Label fraction 10%
None	Supervised baseline [38]	~20	25.6M / 25.6M	25.4	56.4	48.4	80.4
	Pseudo-label [19, 38]	~100	25.6M / 25.6M	-	-	51.6	82.4
	VAT+EntMin. [26, 12, 38]	-	25.6M / 25.6M	-	68.8	-	88.5
	S4L-Rotation [38]	~200	25.6M / 25.6M	-	53.4	-	83.8
	UDA (RandAug) [36]	-	25.6M / 25.6M	-	68.8	-	88.5
	FixMatch (RandAug) [32]	~300	25.6M / 25.6M	-	71.5	-	89.1
	FixMatch w. DA	~400	25.6M / 25.6M	53.4	70.8	74.4	89.0
	CoMatch	~400	30.0M / 25.6M	**66.0**	**73.6**	**86.4**	**91.6**
PIRL [25]	Fine-tune	~800	26.1M / 25.6M	30.7	60.4	57.2	83.8
PCL [21]		~200	25.8M / 25.6M	-	-	75.3	85.6
SimCLR [5]		~1000	30.0M / 25.6M	48.3	65.6	75.5	87.8
BYOL [13]		~1000	37.1M / 25.6M	53.2	68.8	78.4	89.0
SwAV [3]		~800	30.4M / 25.6M	53.9	70.2	78.5	89.9
MoCov2 [7]	Fine-tune	~800	30.0M / 25.6M	49.8	66.1	77.2	87.9
	FixMatch w. DA	~1200	30.0M / 25.6M	59.9	72.2	79.8	89.5
	CoMatch	~1200	30.0M / 25.6M	**67.1**	**73.7**	**87.1**	**91.4**
SimCLRv2* [6]	Fine-tune	~800	34.2M / 29.8M	57.9	68.4	82.5	89.2
	Teacher distillation	~2400	829.2M / 29.8M	73.9	77.5	91.5	93.4

Table 2: Accuracy for ImageNet with 1% and 10% of labeled examples. SimCLRv2* [6] uses larger models for training and test.

- ImageNet ILSVRC-2012: ResNet-50

Experiments

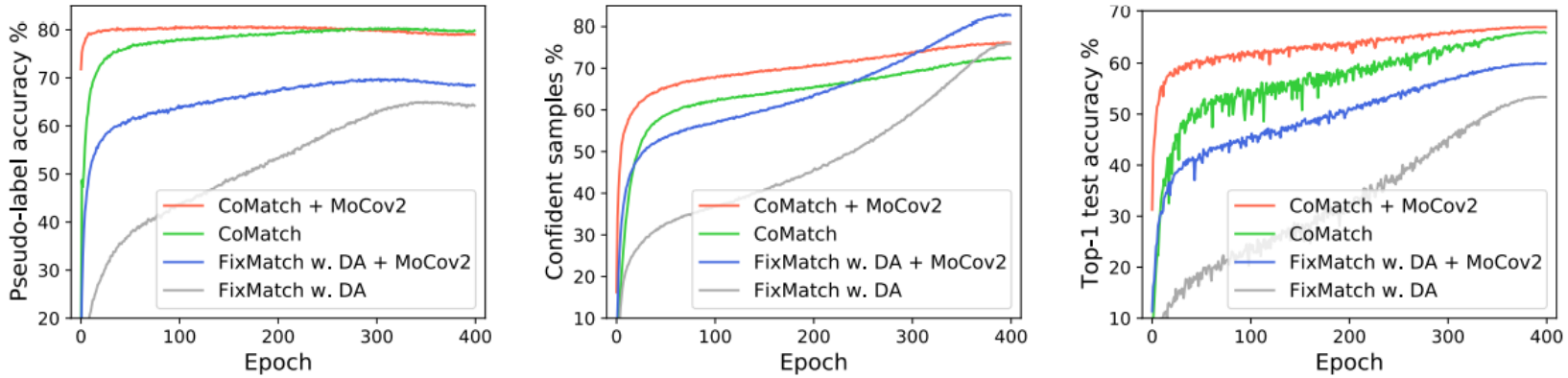


Figure 3: Plots of different methods as training progresses on ImageNet with 1% labels. (a) Accuracy of the confident pseudo-labels *w.r.t* to the ground-truth labels of the unlabeled samples. (b) Ratio of the unlabeled samples with confident pseudo-labels that are included in the unsupervised classification loss. (3) Top-1 accuracy on the test data.

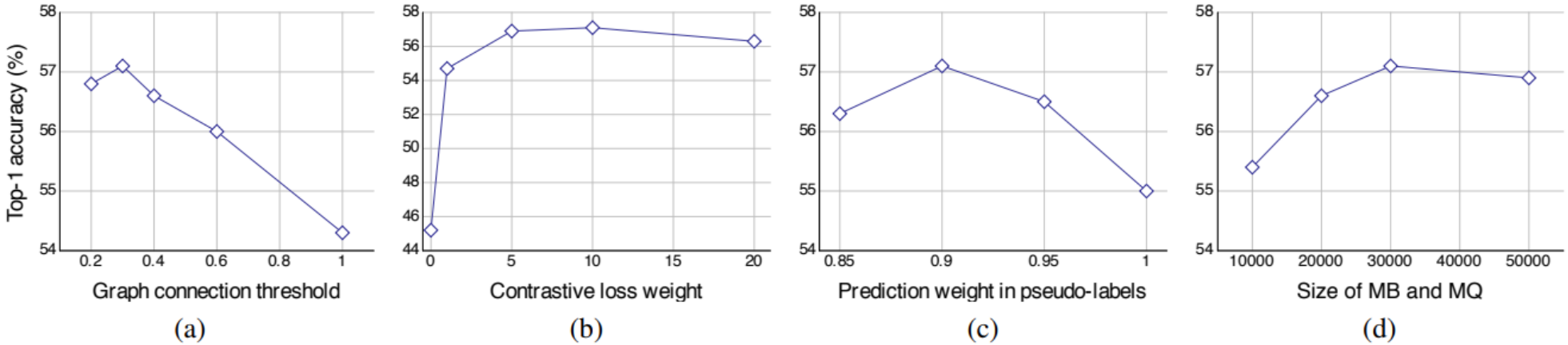


Figure 4: Plots of ablation studies on CoMatch. The default hyperparameter setting achieves 57.1% (ImageNet with 1% labels, trained for 100 epochs). FixMatch with EMA pseudo-label achieves 43.9%. (a) Varying the threshold T which controls the sparsity of edges in the pseudo-label graph. $T = 1$ reduces to self-supervised contrastive learning. (b) Varying the weight λ_{ctr} for the contrastive loss. $\lambda_{ctr} = 0$ removes contrastive learning. (c) Varying α, the weight of the EMA model's prediction in generating pseudo-labels. $\alpha = 1$ reduces to pseudo-labeling with mean teacher [33]. (d) Varying K, the number of samples in both the memory bank and the momentum queue.

Experiments

- **Transfer of Learned Representations**

The number of samples per-class (k) in the downstream datasets

PASCAL VOC2007 object classification and Places205 for scene recognition

Method	#ImageNet labels	#Pre-train epochs	k=4	k=8	k=16	k=64	Full
Supervised	100%	90	73.51±2.12	79.60±0.61	82.75±0.34	85.55±0.12	87.12
MoCov2 [7]	0%	800	70.47±2.18	76.74±0.87	80.61±0.53	84.60±0.11	86.83
SwAV [3]		400	68.04±2.39	75.06±0.73	79.46±0.55	84.24±0.13	86.86
MoCov2 [7]	1%	800	71.82±2.09	77.35±0.83	81.33±0.50	84.98±0.14	87.05
CoMatch		400	72.81±1.50	79.18±0.51	82.30±0.46	85.65±0.17	87.66
MoCov2 [7]	10%	800	73.09±2.02	79.37±0.40	82.05±0.46	85.41±0.16	87.48
CoMatch		400	**74.56**±2.04	**80.60**±0.31	**83.24**±0.43	**86.07**±0.16	**87.91**

(a) VOC07

Method	#ImageNet labels	#Pre-train epochs	k=4	k=8	k=16	k=64	k=256
Supervised	100%	90	27.20±0.41	32.08±0.45	35.95±0.21	41.81±0.17	45.74±0.14
MoCov2 [7]	0%	800	25.34±0.51	30.64±0.39	35.08±0.34	42.18±0.10	46.96±0.06
SwAV [3]		400	25.32±0.46	31.00±0.47	35.65±0.28	42.60±0.11	**47.51**±0.20
MoCov2 [7]	1%	800	26.22±0.50	31.33±0.40	35.55±0.35	42.20±0.11	46.95±0.07
CoMatch		400	27.15±0.42	32.36±0.37	36.56±0.33	42.97±0.11	47.32±0.18
MoCov2 [7]	10%	800	27.19±0.47	32.11±0.49	36.00±0.30	42.31±0.13	46.88±0.08
CoMatch		400	**28.11**±0.33	**33.05**±0.46	**36.98**±0.28	**43.06**±0.22	47.10±0.11

(b) Places

Table 3: Linear classification on VOC07 and Places using models pre-trained on ImageNet. We vary the number of examples per-class (k) on the down-stream datasets. We report the average result with std across 5 runs.

Thanks