MarginMatch: Improving Semi-Supervised Learning with Pseudo-Margins

Tiberiu Sosea Cornelia Caragea
University of Illinois Chicago
tsosea2@uic.edu cornelia@uic.edu

Background

What's semi-supervised learning?

Few labeled data

+

Numerous unlabeled data

Background

Development of Different SSL Methods

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

Motivation

How to make unlabeled sample selection more adaptive?

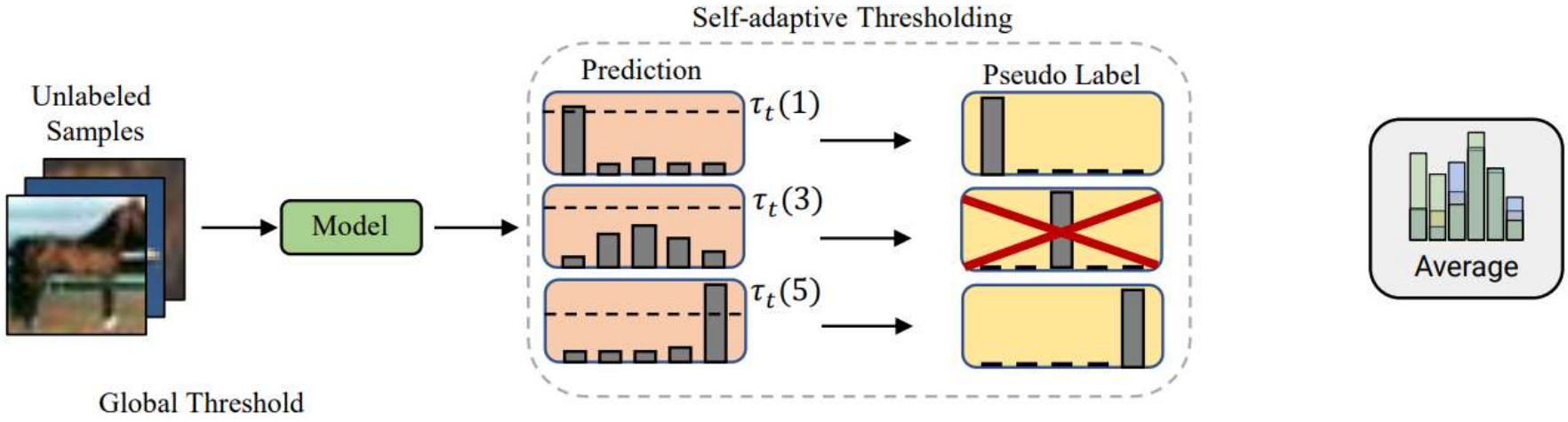
Figure 1. Incorrect pseudo-labels propagated until the end of the training process for FixMatch and FlexMatch on ImageNet.

Confirmation Bias
If the initial predictions are **biased** or **contain errors**, the model would tend to be **overconfident** in its incorrect predictions.

Overconfident
Even when the **high-confidence threshold** is used in FixMatch can result in wrong pseudo-labels.

Motivation

One Strike and you're out?



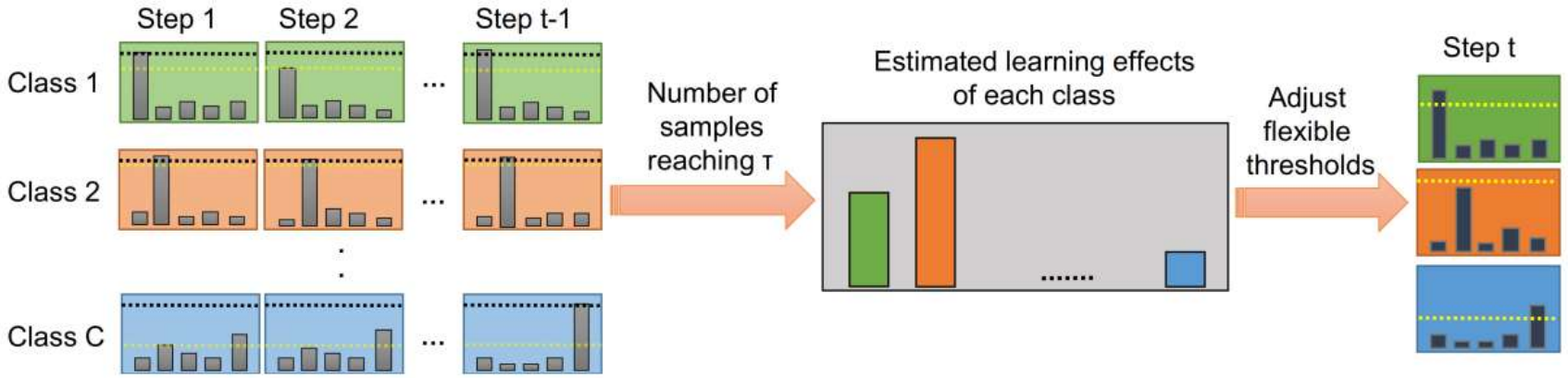
Instead of using only the model's confidence on an unlabeled example at **an arbitrary iteration** to decide if the example should be masked or not, MarginMatch **analyzes the behavior** of the model on the pseudo-labeled examples **as the training progresses**, to ensure **low quality predictions** are masked out.

Instead of using ***One Strike and you're out*** rule

How to measure prediction quality more accurately?

Prerequisite

FlexMatch



$$\sigma_t(c) = \sum_{n=1}^{N} \mathbb{1}(\max(p_{m,t}(y|u_n)) > \tau) \cdot \mathbb{1}(\arg\max(p_{m,t}(y|u_n) = c).$$

Reflects the learning effect of class c at time step t

$$\mathcal{L}_{u,t} = \frac{1}{\mu B} \sum_{b=1}^{\mu B} \mathbb{1}(\max(q_b) > \mathcal{T}_t(\arg\max(q_b)))H(\hat{q}_b, p_m(y|\Omega(u_b))),$$

$$\beta_t(c) = \frac{\sigma_t(c)}{\max_c \sigma_t},$$

$$\mathcal{T}_t(c) = \beta_t(c) \cdot \tau.$$

Method

Instead of using only the model's **current belief**, MarginMatch monitors the training dynamics of unlabeled data over the iterations by investigating the margins:

$$\mathrm{PM}_c^t(\hat{x}) = z_c - max_{c!=i}(z_i)$$

We average all the margins with respect to c from the first iteration until t and obtain the average pseudo-margin (APM) as follows:

$$\mathrm{APM}_c^t(\hat{x}) = \frac{1}{t}\sum_{j=1}^{t}\mathrm{PM}_c^j(\hat{x})$$

If over the iterations, the model predictions do **not agree frequently** with the pseudo-label c from iteration t on the predicted label, the APM for class c will have a **low** value.

Old pseudo-margins eventually become deprecated due to the **large** number of **unlabeled data**, so we use EMA to place **more importance** on **recent iterations**.

$$\mathrm{APM}_c^t(\hat{x}) = \mathrm{PM}_c^t(\hat{x}) * \frac{\delta}{1+t} + \mathrm{APM}_c^{t-1}(\hat{x}) * (1 - \frac{\delta}{1+t})$$

Method

Illustration of Pseudo-margin (PM) Metric

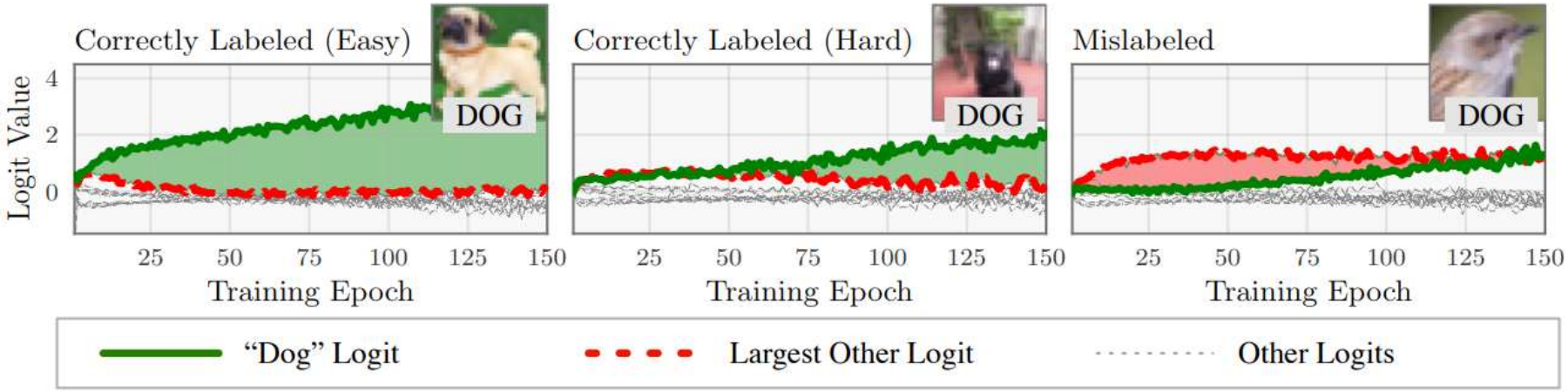


Figure 2: Illustration of the *Area Under the Margin* (AUM) metric. The graphs display logit trajectories for easy-to-learn dogs (left), hard-to-learn dogs (middle), and BIRDS mislabeled as DOGS (right). (Each plot's logits are averaged from 50 CIFAR10 training samples, 40% label noise.) AUM is the shaded region between the DOG logit and the largest other logit. Green/red regions represent positive/negative AUM. Correctly-labeled samples have larger AUMs than mislabeled samples.

Identifying Mislabeled Data using the Area Under the Margin Ranking

Method

The margin at epoch t captures how much larger the (potentially incorrect) assigned logit is than all other logits:

$$M^{(t)}(\mathbf{x}, y) = \overbrace{z_y^{(t)}(\mathbf{x})}^{\text{assigned logit}} - \overbrace{\max_{i \neq y} z_i^{(t)}(\mathbf{x})}^{\text{largest other logit}}.$$

Average a sample's margin measured at each training epoch, a metric we refer to as area under the margin (AUM):

$$\text{AUM}(\mathbf{x}, y) = \frac{1}{T} \sum_{t=1}^{T} M^{(t)}(\mathbf{x}, y),$$

APM(AUM) FlexMatch

$$\mathcal{L}_u = \sum_{i=1}^{\nu B} 1(\text{AM}^t_{\hat{p}_\theta(y|\pi(\hat{x}_i))}(\hat{x}_i) > \gamma^t) 1(\max(p_\theta(y|\pi(\hat{x}_i))) > \mathcal{T}^t_{\hat{p}_\theta(y|\pi(\hat{x}_i))}) H(\hat{p}_\theta(y|\pi(\hat{x}_i)), p_\theta(y|\Pi(\hat{x}_i)))$$

How to Estimate APM(AUM) Threshold?

Method

Mislabeled Data Can Be Identified by Setting a Threshold

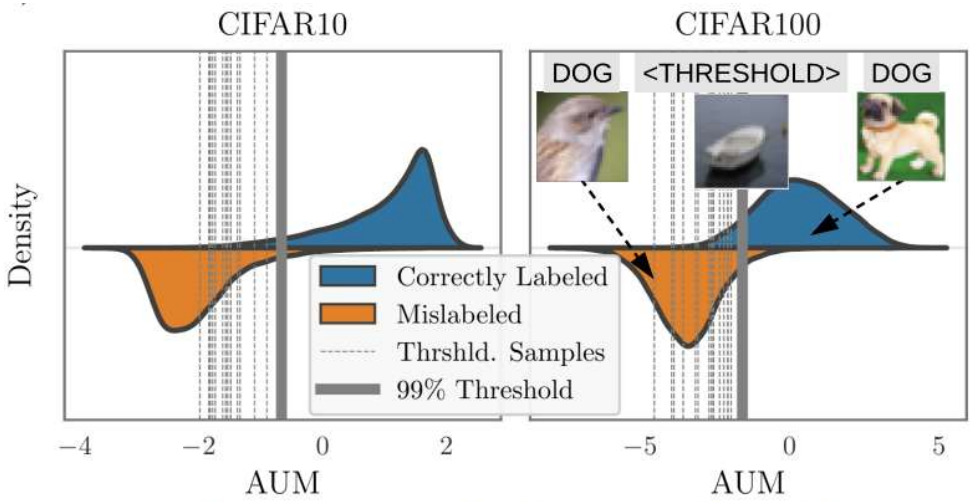


Figure 3: Illustrating the role of *threshold samples* on CIFAR10/100 with 40% mislabeled samples. Histograms of AUMs for correctly-labeled (blue) and mislabeled samples (orange). Dashed lines represent the AUM values of threshold samples. The 99th percentile of threshold AUMs (solid gray line) separates correctly- and mislabeled data.

With access to a trusted **validation set**, a threshold can be learned through a hyperparameter sweep.

But how to learn a threshold without validation data?

Randomly sample a subset of unlabeled examples from U to create **erroneous(threshold)** examples E, which are assigned to an inexistent (or virtual) class C + 1.

Compute APM^t_{C+1} to choose the APM threshold γ^t

$$\mathcal{L}_e = \sum_{i=1}^{B} H(C+1, p_\theta(y|\Pi(\tilde{x}_i)))$$

Strong Augmentation

$$\mathcal{L} = \mathcal{L}_s + \lambda(\mathcal{L}_u + \mathcal{L}_e)$$

Method

Algorithm 1 MarginMatch

Require: Labeled data L; unlabeled data U; erroneous examples E; maximum number of iterations T; number of classes $C+1$ (C original classes plus one virtual class of erroneous examples); θ model; π weak augmentations; Π strong augmentations.

Initialize the Average Pseudo-Margin (APM) threshold γ^1 at the first iteration to a small value (e.g., $\gamma^1 = -\infty$).
for $t = 1$ to T **do**
  Estimate learning status α_c (using Eq. 2) and calculate the class-wise flexible thresholds $\mathcal{T}_c^t$ (using Eq. 3) for each class c.
  while U not exhausted **do**
    Labeled batch $L_b = \{(x_1, y_1), ..., (x_B, y_B)\}$, unlabeled batch $U_b = \{\hat{x}_1, ..., \hat{x}_{\nu B}\}$, erroneous (or mislabeled) batch $E_b = \{(\tilde{x}_1, C+1), ..., (\tilde{x}_B, C+1)\}$
    for $x \in U_b \cup E_b$ **do**
      Compute logits z_c for each class c after applying weak augmentations when $x \in U_b$ and strong augmentations when $x \in E_b$.
      Calculate pseudo-margin PM_c^t (using Eq. 5) and update Average PM_c^t (using Eq. 6) for each $c = 1$ to $C+1$.
    end for
    Minimize $\mathcal{L} = \mathcal{L}_s + \lambda(\mathcal{L}_u + \mathcal{L}_e)$
      $\mathcal{L}_s = \frac{1}{B}\sum_{i=1}^{B} H(y_i, p_\theta(y|\pi(x_i)))$
      $\mathcal{L}_u = \sum_{i=1}^{\nu B} \mathbb{1}(\mathrm{AM}^t_{\hat{p}_\theta(y|\pi(\hat{x}_i))}(\hat{x}_i) > \gamma^t) \times \mathbb{1}(\max(p_\theta(y|\pi(\hat{x}_i))) > \mathcal{T}^t_{\hat{p}_\theta(y|\pi(\hat{x}_i))}) \times H(\hat{p}_\theta(y|\pi(\hat{x}_i)), p_\theta(y|\Pi(\hat{x}_i)))$
      $\mathcal{L}_e = \sum_{i=1}^{B} H(C+1, p_\theta(y|\Pi(\tilde{x}_i)))$
  end while
  Update γ^{t+1} as the 95^{th} percentile erroneous sample APM^t_{C+1}.
end for

Experiment

Dataset	CIFAR-10			CIFAR-100			SVHN			STL-10		
#Labels/Class	4	25	400	4	25	100	4	25	100	4	25	100
Pseuso-Labeling	$74.61_{0.26}$	$46.49_{2.20}$	$15.08_{0.19}$	$87.45_{0.85}$	$57.74_{0.28}$	$36.55_{0.24}$	$64.61_{5.60}$	$25.21_{2.03}$	$9.40_{0.32}$	$74.68_{0.99}$	$55.45_{2.43}$	$32.64_{0.71}$
UDA	$10.79_{3.75}$	$5.32_{0.06}$	$4.41_{0.07}$	$48.95_{1.59}$	$29.43_{0.21}$	$23.87_{0.23}$	$5.34_{4.27}$	$4.26_{0.39}$	$1.95_{0.01}$	$37.82_{8.44}$	$9.81_{1.15}$	$6.81_{0.17}$
MixMatch	$45.24_{2.15}$	$12.76_{1.14}$	$7.13_{0.34}$	$62.15_{2.17}$	$41.51_{1.19}$	$28.16_{0.24}$	$46.18_{1.78}$	$3.98_{0.17}$	$3.5_{0.13}$	$34.15_{1.54}$	$8.95_{0.32}$	$10.41_{0.73}$
ReMixMatch	$5.27_{0.19}$	$4.85_{0.13}$	$4.04_{0.12}$	$47.15_{0.76}$	$27.14_{0.23}$	$23.78_{0.12}$	$4.23_{0.31}$	$3.18_{0.04}$	$1.94_{0.06}$	$31.51_{0.75}$	$8.54_{0.48}$	$6.19_{0.24}$
FixMatch	$7.8_{0.28}$	$4.91_{0.05}$	$4.25_{0.08}$	$48.21_{0.82}$	$29.45_{0.16}$	$22.89_{0.12}$	$3.97_{1.18}$	**$3.13_{1.03}$**	$1.97_{0.03}$	$38.43_{4.14}$	$10.45_{1.04}$	$6.43_{0.33}$
FlexMatch	$5.04_{0.06}$	$5.04_{0.09}$	$4.19_{0.01}$	$39.99_{1.62}$	$26.96_{0.08}$	$22.44_{0.15}$	$8.19_{3.20}$	$7.78_{2.55}$	$6.72_{0.30}$	$29.15_{1.32}$	$8.23_{0.13}$	$5.77_{0.12}$
MarginMatch	**$4.91_{0.07}$**	**$4.73_{0.12}$**	**$3.98_{0.02}$**	**$36.97_{1.32}$**	**$23.71_{0.13}$**	**$21.39_{0.12}$**	**$3.75_{1.20}$**	$3.14_{1.17}$	**$1.93_{0.01}$**	**$25.37_{3.58}$**	**$7.31_{0.35}$**	**$5.52_{0.15}$**

Table 1. Test error rates on CIFAR-10, CIFAR-100, SVHN, and STL-10 datasets. Best results are shown in **blue**.

Experiment

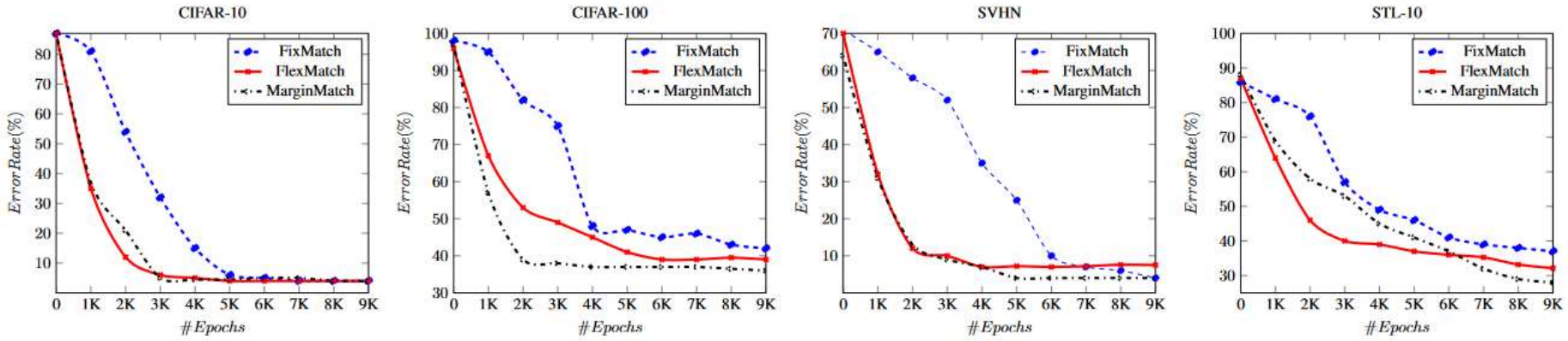


Figure 2. Convergence speed of MarginMatch against FixMatch and FlexMatch with 4 labels per class.

Dataset	ImageNet		WebVision	
	TOP-1	TOP-5	TOP-1	TOP-5
Supervised	48.39	25.49	49.58	26.78
FixMatch	43.66	21.80	44.76	22.65
FlexMatch	42.02	19.49	43.87	22.07
MarginMatch	**41.05**	**18.28**	**43.08**	**21.13**

Table 2. Test error rates on the ImageNet and WebVision datasets. Best results are shown in **blue**.

Experiment

Ablation Study

δ	0.95	0.99	0.995	0.997	0.999	1
ERR RATE	38.13	38.05	37.92	**37.91**	39.12	39.72

EMA coefficient

Table 3. Error rates obtained on CIFAR-100 with four examples per class and various smoothing values δ. Best result is in **blue**.

Dataset	CIFAR-10			CIFAR-100			SVHN			STL-10		
#Labels/Class	4	25	400	4	25	100	4	25	100	4	25	100
Avg Confidence	$23.87_{2.73}$	$14.21_{1.37}$	$7.54_{0.78}$	$41.23_{2.15}$	$31.49_{1.48}$	$24.11_{2.36}$	$8.99_{4.27}$	$6.54_{0.39}$	$4.73_{0.01}$	$31.67_{8.44}$	$14.87_{1.15}$	$7.59_{0.17}$
Avg Entropy	$8.58_{0.41}$	$6.18_{0.15}$	$5.85_{0.12}$	$45.10_{0.91}$	$26.02_{1.11}$	$22.13_{0.25}$	$15.69_{1.25}$	$12.74_{0.78}$	$9.33_{0.05}$	$29.54_{3.51}$	$10.63_{1.35}$	$10.84_{0.47}$
Avg Margin	$7.25_{0.29}$	$5.38_{0.76}$	$4.73_{0.09}$	$39.72_{1.52}$	$25.21_{0.52}$	$23.18_{0.17}$	$18.45_{1.36}$	$11.29_{0.93}$	$8.40_{0.04}$	$28.45_{4.28}$	$9.34_{1.34}$	$7.59_{0.21}$
EMA Confidence	$4.91_{0.45}$	$4.74_{0.09}$	$3.99_{0.06}$	$38.67_{0.74}$	$25.61_{0.12}$	$21.48_{0.17}$	$3.84_{0.23}$	$3.25_{0.03}$	$1.93_{0.09}$	$25.9_{0.81}$	$7.6_{0.42}$	$5.74_{0.57}$
EMA Entropy	$6.4_{0.43}$	$8.34_{0.12}$	$4.21_{0.09}$	$41.63_{0.76}$	$36.84_{0.13}$	$22.52_{0.07}$	$3.81_{1.26}$	$3.17_{0.87}$	$2.14_{0.04}$	$27.21_{4.05}$	$8.28_{1.01}$	$6.79_{0.27}$
EMA Margin	**$4.91_{0.07}$**	**$4.73_{0.12}$**	**$3.98_{0.02}$**	**$36.97_{1.32}$**	**$23.71_{0.13}$**	**$21.39_{0.12}$**	**$3.75_{1.20}$**	**$3.14_{1.17}$**	**$1.93_{0.01}$**	**$25.37_{3.58}$**	**$7.31_{0.35}$**	**$5.52_{0.15}$**

Table 4. Test error rates comparing pseudo-margin with confidence and entropy. Best results are shown in **blue**.

APM(AUM)

FlexMatch

$$\mathcal{L}_u = \sum_{i=1}^{\nu B} 1\left(\mathrm{AM}^t_{\hat{p}_\theta(y|\pi(\hat{x}_i))}(\hat{x}_i) > \gamma^t\right) 1\left(\max(p_\theta(y|\pi(\hat{x}_i))) > \mathcal{T}^t_{\hat{p}_\theta(y|\pi(\hat{x}_i))}\right) H\left(\hat{p}_\theta(y|\pi(\hat{x}_i)), p_\theta(y|\Pi(\hat{x}_i))\right)$$

Experiment

Analysis

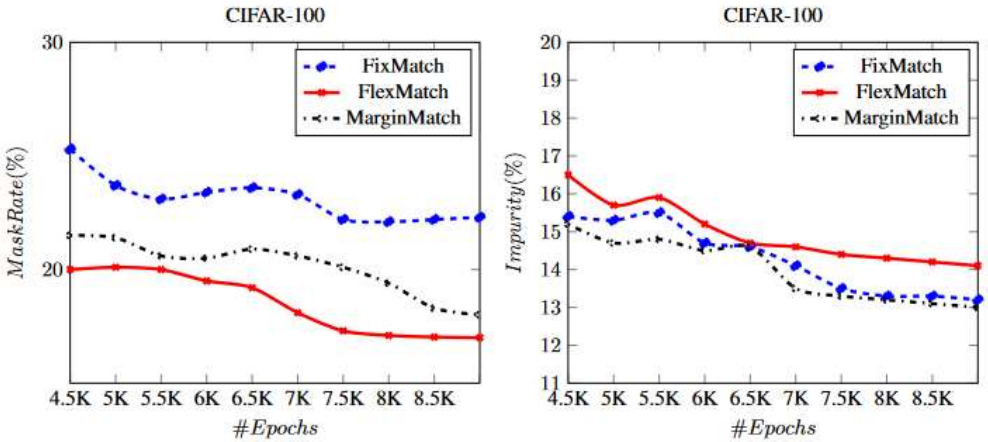


Figure 3. Mask rate and impurity on CIFAR-100 with 4 labeled examples per class.

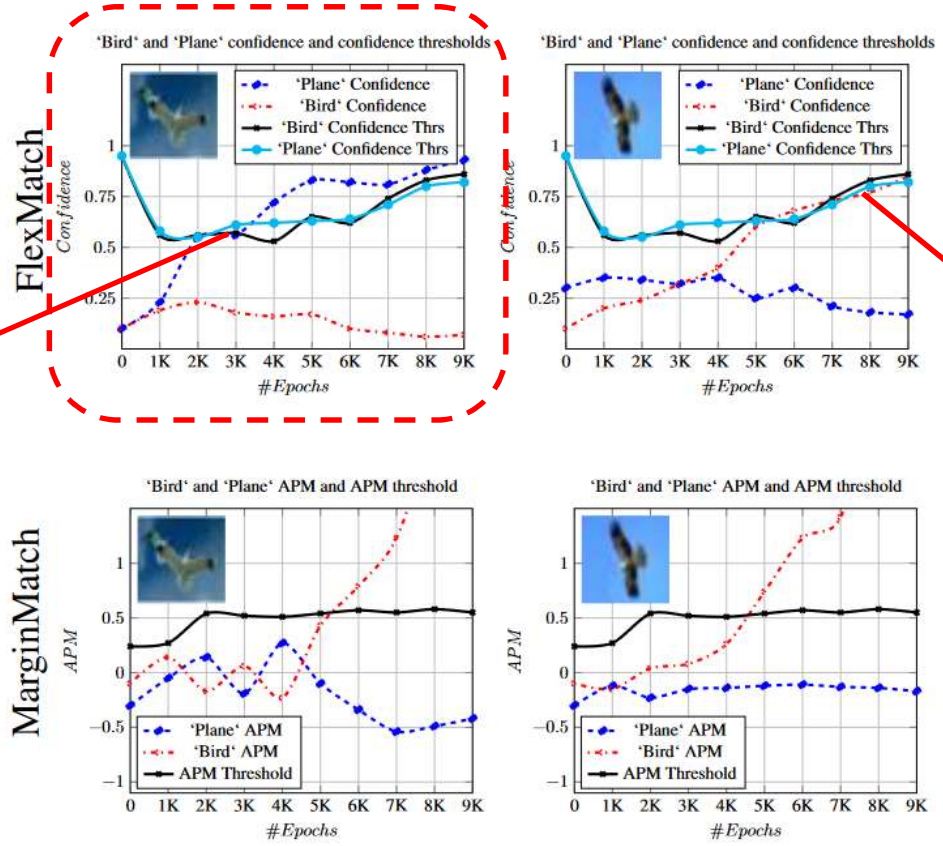
Mask rate is defined as the fraction of pseudo-labeled examples that do **not participate in** the training at epoch t due to confidence masking or pseudo-margin masking (or both).

Impurity in contrast is defined as the fraction of pseudolabeled examples that do **participate in** the training at epoch t **but with a wrong label.**

Experiment

Analysis

Oops, FlexMatch made a **mistake** here. Eventually this sample is wrongly classified.



FlexMatch is still **not** quite **confident** of its prediction at the end of the training.

Two bird images from CIFAR-10. Because of the **resemble** characteristics(background) between birds and plane, the model will get **confused**.