

PAPER REPRODUCIBILITY CHALLENGE
FREEMATCH: SELF-ADAPTIVE CLUSTERING FOR SEMI-SUPERVISED LEARNING

Shreejal Trivedi

[EECS 6322] - Neural Networks and Deep Learning
York University

April 16, 2023

TABLE OF CONTENT

1	Disclaimer	2
2	Motivation	3
3	Related Work	4
4	Make Takeaways from Paper	5
4.1	Definition of Semi-Supervised Learning	6
4.2	Self-Adaptive Thresholding (SAT)	7
4.3	Self Adaptive Fairness (SAF)	9
5	Implementation	11
6	Results	12
6.1	Accuracy on CIFAR10	12
6.2	Precision, Recall, and F1 Score on CIFAR10	13
7	Conclusion	15
8	References	16

DISCLAIMER

- ▶ All the theorems, propositions, and proof are taken from the paper by Wang, Chen, Heng, et al. 2023. I have just reproduced the paper to show the main experiments and the results following the propositions of their work in Semi-Supervised Learning.
- ▶ I would like to thank the authors of Wang, Chen, Heng, et al. 2023 for their outstanding work on a new approach to semi-supervised learning and detailed analysis of the working of the same. To get into the details of all the loss functions and their proofs, read the original paper *FreeMatch: Self-adaptive Thresholding for Semi-Supervised Learning*

MOTIVATION

- ▶ Semi-Supervised Learning is one of the emerging fields in deep learning due to its direct correspondence to the real-world deployed systems, as the number of labeled data for custom use-case systems might not be present in abundance and the data might be skewed.
- ▶ Semi-supervised learning is also useful in the scalability of the deployed systems, which can consider the site-specific training strategies with every few labeled data for the particular site.
- ▶ Apart from image classification, SSL is drastically making its impact in other computer vision applications such as object detection, tracking, and segmentation. This project aimed to reproduce the important results claimed by the authors of the paper.
- ▶ Following results and experiments were done by taking into account the paper **FreeMatch: Self-adaptive Thresholding for Semi-supervised learning**[Wang, Chen, Heng, et al. 2023].
- ▶ They proposed two algorithms viz. *Self Adaptive Thresholding* and *Self Adaptive Fairness Regularization* helped them to give state-of-the-art results on the standard image classification datasets such as CIFAR10, CIFAR100, SVHN, STL10, and ImageNet.

RELATED WORK

- ▶ Recently, there has been extensive research going into semi-supervised image classification that includes different types of pseudo-labeling and consistency regularization methods to use unlabeled data efficiently during the training of CNNs.
- ▶ But most papers like FixMatch[Sohn et al. 2020], ReMixMatch[Berthelot, Carlini, et al. 2020], UDA[Xie et al. 2020] use fixed high thresholds for pseudo-labeling tasks.
- ▶ This makes the training unstable as different classes might need different thresholds based on the hardness and the distribution of the particular class in the dataset and neglects many training examples at the beginning of the training.
- ▶ To counter this argument, methods like AdaMatch [Berthelot, Roelofs, et al. 2022] use a warm-up strategy to increase the global threshold so that most unlabelled data is used during the early phase of the training. Also, FlexMatch [Zhang et al. 2022] uses the local threshold learning for each class adaptively based on the confidence of the network output.
- ▶ **This makes us wonder about adjusting both the local threshold and global threshold simultaneously in the learning process so that the maximum number of unlabeled data is utilized and, at the same time, the skewness constraint, as well as the hardness(in terms of learning) of class, is maintained in the loss propagation.**

MAKE TAKEAWAYS FROM PAPER

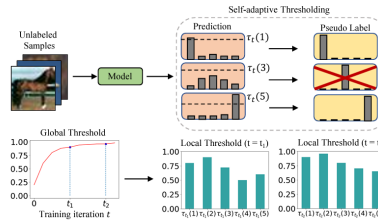


Figure. 1. Taken from Wang, Chen, Heng, et al. 2023

- ▶ Wang, Chen, Heng, et al. 2023 came up with an interesting approach of **Self Adaptive Thresholding(SAT)** to adaptively learn local-global thresholds of class and dataset, respectively.
- ▶ This helps the authors to differentiate the intra and inter-class discrepancies. They also proposed a **Self Adaptive Class Fairness Regularization(SAF)** function to handle barely supervised settings.
- ▶ It eventually normalizes the batch's distribution with the network's marginal class distribution so that the network does not get biased towards the more recurring class during the training. Two main contributions viz. SAT and SAF regularization techniques

MAKE TAKEAWAYS FROM PAPER

SEMI-SUPERVISED LEARNING

Theorem 1 (Formal Definition (Wang, Chen, Heng, et al. 2023))

Let $\mathcal{D}_{\mathcal{L}} = \{(x_n, y_n) : n \in [N_L]\}$ and $\mathcal{D}_{\mathcal{U}} = \{(x_n) : n \in [N_U]\}$ be the set of labeled and unlabeled data respectively. Here, $N_{\mathcal{L}}$ and $N_{\mathcal{U}}$ are the total number of labeled and unlabeled examples. The supervised and consistency loss is given by

$$\mathcal{L}_s = \frac{1}{B} \sum_{b=1}^B \mathcal{H}(y_b, p_m(y|\omega(x_b))) \quad (1)$$

$$\mathcal{L}_u = \frac{1}{\mu B} \sum_{b=1}^{\mu B} \mathbb{1}(\max(q_b) > \tau) \mathcal{H}(\hat{q}_b, Q_b) \quad (2)$$

- ▶ $p_m(y|\omega(x_b))$ is the probability distribution of the model output and $\omega(x_b)$ is the weak augmentation applied to the input image. \mathcal{H} is the cross-entropy function, y_b is the true label, and B is the batch size of the input.
- ▶ μ is the unlabeled ratio, $q_b = p_m(y|\omega(x_b))$, $Q_b = p_m(y|\Omega(x_b))$, $\mathbb{1}(\cdot > \tau)$ acts as an indicator function which masks the loss whose output confidence threshold is greater than τ . $\omega(x_b)$ and $\Omega(x_b)$ are the weak and strong augmentations respectively. Strong augmentations consist of *RandAugment*, *Random Crop*, and *Horizontal Flipping*.

MAKE TAKEAWAYS FROM PAPER

SELF-ADAPTIVE THRESHOLDING (SAT)

- ▶ In semi-supervised learning, the threshold τ plays an important role in selecting the examples for the calculation and stable training.
- ▶ The adaptive global threshold is learned on the overall prediction of the model's output, whereas the local threshold is molded using the local predictions per class.
- ▶ The global threshold is used to modulate the model's confidence in the unlabeled data. Moreover, this threshold will be very less at the start of the training and will gradually increase once the predictions become more confident

$$\tau_t = \begin{cases} \frac{1}{C}, & t = 0 \\ \lambda\tau_{t-1} + (1 - \lambda)\frac{1}{\mu B} \sum_{b=1}^{\mu B} \max(q_b), & \text{otherwise} \end{cases} \quad (3)$$

- ▶ $\lambda \in (0, 1)$ is used as a momentum decay in EMA and $q_b = p_m(y|\omega(x_b))$.
- ▶ Unlike Global SAT, the local thresholding technique uses the softmax output of the logits instead of taking the maximum value. The logic behind this technique is to leverage the discrepancies between the classes in terms of the number of labeled data and the hardness of the class

MAKE TAKEAWAYS FROM PAPER

SELF-ADAPTIVE THRESHOLDING (SAT) CONTD.

- Here, $\tilde{p}_t(c)$ is the average confidence of the model for the class c . Therefore, the final value for the class-wise thresholding is given by

$$\tau_t(c) = \frac{\tilde{p}_t(c)}{\max\{\tilde{p}_t(c) : c \in [C]\}} * \tau_t \quad (4)$$

$$\tilde{p}_t(c) = \begin{cases} \frac{1}{C}, & t = 0 \\ \lambda \tilde{p}_{t-1}(c) + (1 - \lambda) \frac{1}{\mu B} \sum_{b=1}^{\mu B} q_b(c), & \text{otherwise} \end{cases} \quad (5)$$

- At every iteration, the local and global threshold will change. The final consistency loss with the proposed threshold is given by

$$\mathcal{L}_u = \frac{1}{\mu B} \sum_{b=1}^{\mu B} \mathbb{1}(\max(q_b) > \tau_t(\operatorname{argmax}(q_b))) \mathcal{H}(\hat{q}_b, Q_b) \quad (6)$$

MAKE TAKEAWAYS FROM PAPER

SELF ADAPTIVE FAIRNESS

- ▶ The main challenge in a semi-supervised learning setup is when the distribution of the labeled and unlabeled data is skewed and barely supervised (labeled data = 1 per class).
- ▶ The authors tackle this problem by tracking the distribution of the labeled and unlabeled data during the training, which can normalize the skewness of the data during the training.
- ▶ This can help the model not to get biased towards the dominated training class data and help to get over better generalization. This is accomplished by normalizing $\mathbb{E}_{\mu_B}[p_m(y|\Omega(u_b))]$ with the histogram distribution of the pseudo labels.

$$\bar{p} = \frac{1}{\mu_B} \sum_{b=1}^{\mu_B} \mathbb{1}(\max(q_b) > \tau_t(\operatorname{argmax}(q_b))) Q_b \quad (7)$$

$$\bar{h} = \operatorname{Hist}_{\mu_B}(\mathbb{1}(\max(q_b) > \tau_t(\operatorname{argmax}(q_b)))) \hat{Q}_b \quad (8)$$

- ▶ \hat{Q}_b is the pseudo label generated from the strong augmented image and \tilde{h} is the running average of the histogram calculated from weakly augmented pseudo labels.

MAKE TAKEAWAYS FROM PAPER

SELF ADAPTIVE FAIRNESS CONTD.

- ▶ The self-adaptive fairness loss is the cross-entropy between the normalized class distributions of strong and weak augmented outputs, as shown in eq 10

$$\tilde{h} = \lambda \tau_{t-1} + (1 - \lambda) \frac{1}{\mu B} \sum_{b=1}^{\mu B} \text{Hist}_{\mu B}(\hat{q}_b) \quad (9)$$

$$\mathcal{L}_f = -\mathcal{H}(\text{SumNorm}(\frac{\tilde{p}_t}{\tilde{h}_t}), \text{SumNorm}(\frac{\bar{p}_t}{\bar{h}_t})) \quad (10)$$

- ▶ the total loss of training is given by,

$$\mathcal{L} = \mathcal{L}_s + w_u \mathcal{L}_u + w_f \mathcal{L}_f \quad (11)$$

- ▶ Here, w_u and w_f are the hyperparameters that are tuned for different experiments.

IMPLEMENTATION

- The authors conducted the experiments on five datasets viz. CIFAR10, CIFAR100, SVHN, STL10, and ImageNet. Due to GPU constraints, I was only able to finish all the experiments on the **CIFAR10** dataset and was successful in doing the same.

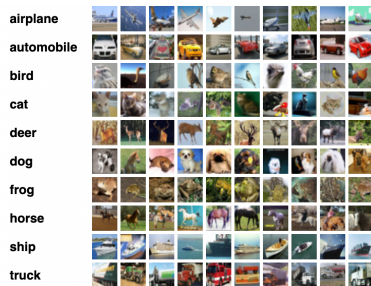


Figure. 2. Classes in CIFAR10 dataset

Specification Type	Name	Value
CIFAR10 Parameters	Batch Size	64
	Unlabeled Ratio	7
	SAT Loss multiplier w_u	1
	SAF loss multiplier w_f	0.03 (0.01 for num labeled=10)
	Image Size	32
Network Parameters	WideResNet	Depth: 28, Width: 2
	Mixed Precision Training	ON
	Weight Decay	5e-4
	Learning Rate	0.03
	SGD Momentum	0.9
	EMA Decay	0.999

Figure. 3. Overall setup of hyperparameters taken from Wang, Chen, Heng, et al. 2023

- The network used was WideResNet-28-2 and it was trained with a learning rate of 0.03, weight decay of 5e-4, and SGD momentum of 0.9.
- The w_u was kept at 0.03 for all the experiments except when the number of labeled examples in CIFAR10 was 10 when 0.01 was used.
- The unlabeled ratio was 7 and the labeled example batch size was 64. The model was trained on one NVIDIA Tesla T4 GPU on Google Cloud.

RESULTS

ACCURACY ON CIFAR 10 DATASET

- ▶ I successfully reproduced the results mentioned in *Table 1, Table 7, Table 8, Table 9* as mentioned in Wang, Chen, Heng, et al. 2023.
- ▶ All the experiments were trained with **Mixed Precision training** on a single Tesla T4 GPU on Google Cloud for around 524288 iterations training steps. The original paper trained the model for 1048576 iterations.

Results	# Labels(10)	# Labels(40)	# Labels(250)	# Labels(4000)
FreeMatch	92.93 ± 4.24	95.10 ± 0.04	95.12 ± 0.18	95.9 ± 0.02
Reproduced	93.00	94.13	95.02	95.1

Table. 1 Comparison of accuracy results between the original paper and reproduced paper on CIFAR10 dataset with different numbers of labeled data. **This table refers to Table 1 in Wang, Chen, Heng, et al. 2023**

RESULTS

PRECISION RECALL AND F1 SCORE ON CIFAR10 DATASET

- As seen in Table 1 and Table 2 the reproduced results are a bit less as all the training was done in mixed precision mode, but the original results posted were done in the normal FP32 training setup.

Method	Metric Name	Value
FreeMatch	Precision	0.8619
	Recall	0.8593
	F1 Score	85.23
	AUC	0.9843
Reproduced	Precision	0.9312
	Recall	93.27
	F1 Score	0.931
	AUC	0.9957

Table. 2. Precision, Recall, F1 Score, and AUC comparison between the methods on CIFAR10(num labeled = 10) . Note that the original paper's authors ran the experiments for different seed values and averaged the results when reporting. The experiments I did were run only once on a random seed value. **This table refers to Table 9 in Wang, Chen, Heng, et al. 2023**

RESULTS

PRECISION RECALL AND F1 SCORE ON CIFAR10 DATASET CONTD.

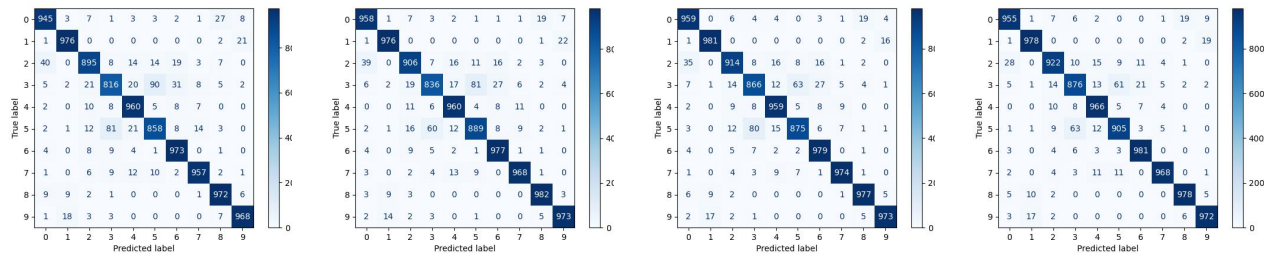









Figure. 4. From Left to Right, Shows the confusion matrix of the CIFAR10 dataset with 10, 40, 250, and 4000 labels, respectively.

- ▶ All the logs for the experiments conducted with tensorboard logs are available here: [Link](#)
- ▶ The repository is located at <https://github.com/shreejalt/freematch-pytorch>

CONCLUSION

- ▶ The experiments on the proposed work by the authors were successfully reproduced on the CIFAR10 dataset.
- ▶ The reproduction of the other three datasets can also be accomplished with proper computing resources.
- ▶ All the experiments were done in the AMP environment of PyTorch with half the number of training iterations which might decrease the accuracy numbers as noted in the tables of this paper, but the numbers were very close to the published results of error rates and other metrics.
- ▶ The link to official repository of FreeMatch: *Link* [Wang, Chen, Fan, et al. 2022]

REFERENCES

-  Berthelot, David, Nicholas Carlini, et al. (2020). *ReMixMatch: Semi-Supervised Learning with Distribution Alignment and Augmentation Anchoring*. arXiv: 1911.09785 [cs.LG].
-  Berthelot, David, Rebecca Roelofs, et al. (2022). *AdaMatch: A Unified Approach to Semi-Supervised Learning and Domain Adaptation*. arXiv: 2106.04732 [cs.LG].
-  Sohn, Kihyuk et al. (2020). *FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence*. arXiv: 2001.07685 [cs.LG].
-  Wang, Yidong, Hao Chen, Yue Fan, et al. (2022). *USB: A Unified Semi-supervised Learning Benchmark for Classification*. arXiv: 2208.07204 [cs.LG].
-  Wang, Yidong, Hao Chen, Qiang Heng, et al. (2023). *FreeMatch: Self-adaptive Thresholding for Semi-supervised Learning*. arXiv: 2205.07246 [cs.LG].
-  Xie, Qizhe et al. (2020). *Unsupervised Data Augmentation for Consistency Training*. arXiv: 1904.12848 [cs.LG].
-  Zhang, Bowen et al. (2022). *FlexMatch: Boosting Semi-Supervised Learning with Curriculum Pseudo Labeling*. arXiv: 2110.08263 [cs.LG].