Appendix for "FlatMatch: Bridging Labeled Data and Unlabeled Data with Cross-Sharpness for Semi-Supervised Learning"

Anonymous Author(s)
Affiliation
Address
email

In this Appendix, we provide additional details and experimental results to complement the proposed method. First, we describe supplementary experimental details in Section 1. Then, we provide extra quantitative results, including a comparison on ImageNet30 [2] dataset, employing FlatMatch to other SSL methods in Section 2. Further, we show more empirical results to qualitatively validate FlatMatch in Section 3. Finally, we summarize this paper and make a discussion on prospective research in Section 4.

1 Supplementary Details

The experimental setting of this paper follows Wang et al. [4]. Specifically, the hyper-parameters are composed of algorithm-dependent parameters and algorithm-independent parameters, which are shown in Table 1 and Table 2, respectively. For algorithm-dependent parameters of FlatMatch, we use the same unlabeled data and labeled data ratio as FreeMatch [5] as well as all other baseline methods to sample data into a mini-batch. The perturbation magnitude α is based on the results from hyper-parameter sensitivity analysis in the main paper and is chosen as 0.05 for all experiments. For updating the historical gradient using a memory buffer, we use EMA with factor α to ensemble the gradient result. Moreover, we choose the thresholding strategy from FreeMatch and use an EMA decay. Note that for combining the cross-sharpness regularization from FlatMatch with empirical risk, we find that there is no need to introduce another weight to trade off the two loss functions, hence the weight for cross-sharpness is just set to 1 for all experiments. For algorithm-independent hyper-parameters, we have listed the important model setting, optimizer parameters, and data sampling setting as below. Note that all baseline methods follow the implementation of USB [4] and are trained with EMA decay with 0.999 to smooth the parameter updating.

Table 1: Algorithm-dependent hyper-parameters.

Algorithm	FlatMatch
Unlabeled Data to Labeled Data Ratio (CIFAR-10/100, STL-10, SVHN)	7
Unlabeled Data to Labeled Data Ratio (ImageNet30)	1
Perturbation magnitude ρ for all experiments	0.05
EMA factor α for updating gradient	0.999
Thresholding EMA decay for all experiments	0.999
Trade-off weight $\lambda_{\text{X-sharp}}$ for cross-sharpness	1

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

Table 2: Algorithm-independent hyper-parameters.

Dataset	CIFAR-10	CIFAR-100	STL-10	SVHN	ImageNet30
Model	WRN-28-2	WRN-28-8	WRN-37-2	WRN-28-2	ResNet-50
Weight decay	5e-4	1e-3	5e-4	5e-4	3e-4
Batch size	64				128
Learning rate	0.03				
SGD momentum	0.9				
EMA decay	0.999				

2 Additional Quantitative Results

In this section, we conduct additional experiments on CIFAR10 and ImageNet30 [2] datasets to compare the performance between some of the most edge-cutting methods, including FixMatch [3], Dash [6], FlexMatch [7], FreeMatch [5], SoftMatch [1], and our FlatMatch.

2.1 Combining FlatMatch with Other Methods on CIFAR10

We choose CIFAR10 dataset with the number of labeled data varied as 40, 250, and 4000, and apply the FlatMatch methodology to several recently proposed SSL methods to show the effectiveness of the proposed cross-sharpness regularization. The results are shown in Table 3, as we can see that our method can further boost the learning performance of all five methods on all three settings, which proves that the cross-sharpness method is quite universal to SSL approaches and can bring non-trivial performance enhancement. Note that in the 40 labels setting, we compute our cross-sharpness on 500 examples with fixed labels, as demonstrated in Section 5.2 from the main paper.

Table 3: Performance on boosting other SSL methods using FlatMatch.

Dataset	CIFAR10		
# label	40	250	4000
FixMatch	7.47±0.28	4.86±0.05	4.21±0.08
FixMatch+FlatMatch	**6.50**±1.25	**4.27**±2.15	**3.92**±1.65
Dash	8.93±3.11	5.16±0.23	4.36±0.11
Dash+FlatMatch	**6.73**±2.49	**4.48**±1.56	**4.02**±1.30
FlexMatch	4.97±0.06	4.98±0.09	4.19±0.01
FlexMatch+FlatMatch	**4.47**±0.92	**4.25**±1.37	**3.88**±0.75
SoftMatch	4.91±0.12	4.82±0.09	4.04±0.02
SoftMatch+FlatMatch	**4.30**±1.32	**3.98**±1.14	**3.84**±0.86
FreeMatch	4.90±0.04	4.88±0.18	4.10±0.02
FlatMatch (from main paper)	**4.28**±1.61	**3.90**±1.72	**3.55**±0.64

2.2 Comparing FlatMatch to Other Methods on ImageNet30

To further testify the performance of FlatMatch on a large-scale dataset, we conduct experiments on ImageNet30 dataset which is a subset from the original ImageNet dataset and contains 30000 training examples with resolution 256×256 from 30 classes. The experiments on ImageNet30 are more time-consuming which normally takes 5 days to finish, much more than CIFAR10 dataset which takes 2 days. We vary the number of labeled data as 1500 and 3000 and show the comparison in Table 4. We observe the effectiveness of FlatMatch over all other baseline methods in both two settings, which again validates the superiority of our method and its effective performance on large-scale datasets.

Table 4: Comparison on ImageNet30.

Dataset	ImageNet30	
# label	1500	3000
FixMatch	12.48±0.67	8.25±0.54
Dash	13.29±1.26	8.79±0.42
FlexMatch	11.48±0.52	8.04±0.75
SoftMatch	10.81±0.40	7.78±0.61
FreeMatch	10.34±0.46	7.21±0.19
FlatMatch	**9.71**±1.55	**6.77**±1.27

3 Additional Qualitative Results

To further evaluate the flatness of different SSL models during training, we leverage a validation set to compute the sharpness. The sharpness is measured by the increase of loss within a ℓ_2 bounded neighbor, which is formally defined as $Sharpness := \mathcal{L}(\theta + \epsilon^*(\theta)) - \mathcal{L}(\theta)$, where $\epsilon^*(\theta) = \arg\max_{\|\epsilon\|_2 \leq \rho} \mathcal{L}(\theta + \epsilon)$. Specifically, we compare the proposed FlatMatch with FixMatch, Dash, and FreeMatch, and use fully supervised learning as a baseline method. The experiments are conducted on CIFAR10 and SVHN datasets whose results are shown in Figure 1. First, we observe that FlatMatch achieves the lowest sharpness curve during training on both two datasets, which indicates the SSL model learned by FlatMatch is more robust to perturbations and would not oscillate significantly when facing changes in parameter space. Moreover, we find that fully supervised learning does not improve the flatness as the training proceeds, while all SSL methods can decrease the sharpness to some extent, which demonstrates that training with unlabeled data can help improve the flatness of SSL models.

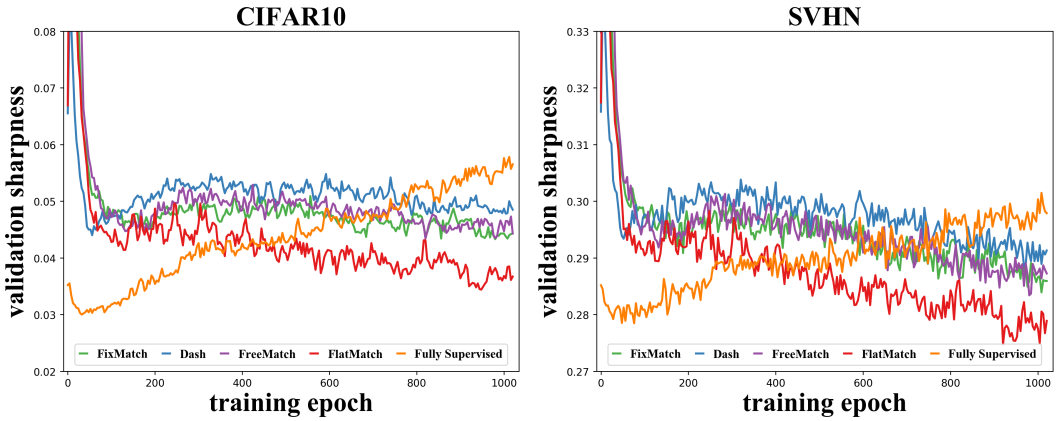

Figure 1: Comparison of sharpness between various SSL methods during training.

Furthermore, as shown in the main paper, we find that FlatMatch has limited performance on extremely scare labeled settings. However, this limitation can be addressed by introducing some unlabeled data with fixed labels to improve the computation of cross-sharpness. Hence, here we investigate the effect of changing the number of fixed on the performance of FlatMatch. Specifically, we conduct experiments on CIFAR10 and SVHN datasets and fixing different numbers of labels as 0 ("w/o fix label"), 250, 500, 1000, 2000, 4000[1]. The results are shown in Figure 2. We find that both too few fixed labels, *i.e.*, 250 labels and too many fixed labels, *i.e.*, 4000 labels in CIFAR10 and 2000 labels in SVHN, would show a performance drop compared to the optimal number, 500 fixed labels. This is because if the number of fixed labels is too small, the gradient computation would be inaccurate, further limiting the learning results. On the other hand, too many fixed labels

[1]The 4000 fixed labels setting is not conducted on SVHN as the performance of 2000 fixed labels setting already shows significantly performance degradation.

would introduce noisy labeled unlabeled data, which would largely mislead the SSL and show serious performance degradation.

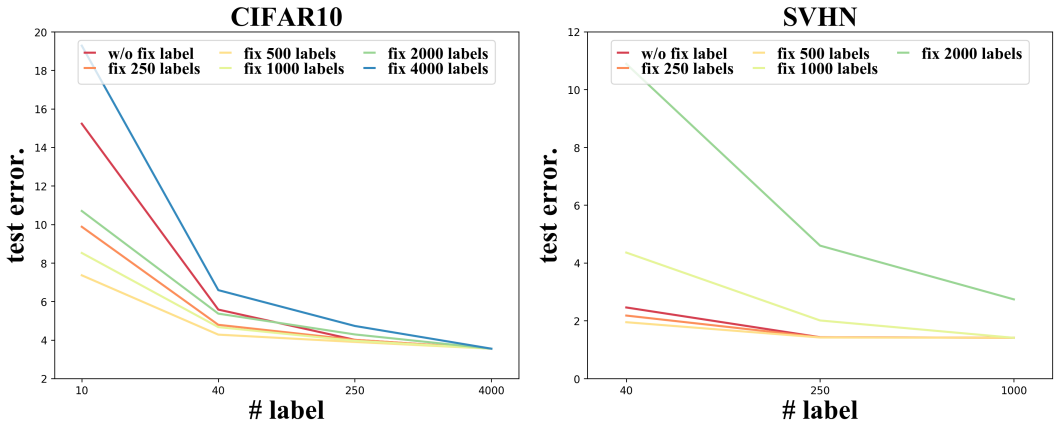

Figure 2: Analysis on changing the number of fixed labels.

4 Summary and Future Work

In this paper, we propose a novel FlatMatch approach that minimizes the cross-sharpness measure to improve the generalization performance of SSL. Through extensive quantitative and qualitative experiments, we have thoroughly evaluated the performance of FlatMatch and demonstrated its superiority to other compared methods. Thanks to the generalization improvement of FlatMatch, the classification accuracy on many scenarios have even passed the fully-supervised baseline.

However, the learning performance of SSL still largely depends on the careful selection of labeled data. Specifically, in the barely-supervised learning scenario, if the selected scarce labeled data deviate from the cluster center, the learning performance of many existing SSL methods would be significantly affected. This is due to the generalization performance between labeled data and unlabeled data being largely mismatched. Under this scenario, the performance of FlatMatch should be further evaluated.

References

[1] Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. In *ICLR*, 2023.

[2] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[3] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020.

[4] Yidong Wang, Hao Chen, Yue Fan, Wang Sun, Ran Tao, Wenxin Hou, Renjie Wang, Linyi Yang, Zhi Zhou, Lan-Zhe Guo, et al. Usb: A unified semi-supervised learning benchmark for classification. In *NeurIPS*, volume 35, pages 3938–3961, 2022.

[5] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Zhen Wu, and Jindong Wang. Freematch: Self-adaptive thresholding for semi-supervised learning. In *ICLR*, 2023.

[6] Yi Xu, Lei Shang, Jinxing Ye, Qi Qian, Yu-Feng Li, Baigui Sun, Hao Li, and Rong Jin. Dash: Semi-supervised learning with dynamic thresholding. In *ICML*, pages 11525–11536. PMLR, 2021.

[7] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In *NeurIPS*, volume 34, pages 18408–18419, 2021.