# CaliMatch: Adaptive Calibration for Improving Safe Semi-supervised Learning

## Supplementary Material

## S-1. Detailed Experiment Setups

**Datasets.** Table S-1 summarizes the experimental settings of five datasets in this study under realistic SSL evaluation protocols [8] with label distribution mismatch scenarios. When we evaluated the SSL methods on multiclass classification in Tables 1 and 2 of the main paper, and Table S-3, we used only testing samples whose class labels align with the label distribution of the labeled dataset. On the other hand, when we evaluated the SSL methods on unseen-label data detection in Table 3 of the main paper and Table S-4, we used the original testing datasets, which contain all classes, including classes not seen from the label distribution of the labeled dataset. When we set the percentage of unseen-label data existence ($\kappa$) to 60% in the unlabeled dataset, it indicates that 60% of the training unlabeled samples come from unseen classes, while the rest of the samples come from seen classes.

**Implementation for CIFAR-10, CIFAR-100, SVHN, and TinyImageNet.** Table S-2 summarizes the implementation details and hyperparameters used in our experiments on CIFAR-10, CIFAR-100, SVHN, and TinyImageNet. All hyperparameters listed in Table S-2 were chosen based on achieving the highest average validation accuracy in multiclass classification or were adapted from previous studies. In the validation stage, we used 10% of the training dataset for each of the four datasets. We used standard normalization for scaling image data and several image augmentations, such as weak and strong augmentations, to implement consistency regularization on SSL methods. The weak augmentation used in this study includes horizontal-flip and random-crop, and the strong augmentation is grounded on RandAugment [2]. Additionally, when implementing other SSL methods for comparison with CaliMatch in our experimental setup, we referenced the official implementations provided by the authors: MTC [13], FixMatch [11], OpenMatch [9], IOMatch [5], SCOMatch [12], and ADELLO [10]. To the best of our knowledge, there is no official code for the SafeStudent method; therefore, we directly implemented it on our experimental setup, and the source codes for SafeStudent and CaliMatch are distributed on our GitHub*. All experiments were conducted using a single 12-GB GPU, such as an NVIDIA 3080 Ti.

**Implementation for ImageNet.** For our ImageNet experiments, we trained CaliMatch, OpenMatch, and a supervised learning baseline using a distributed data-parallel approach on two NVIDIA A100 GPUs (80 GB) with 32 CPU cores. The backbone architecture is ResNet 50 [3], and training was conducted with a batch size of 160 labeled samples and 640 unlabeled samples per iteration. The models were trained for 200 epochs, following a learning rate schedule that included a linear warm-up phase over the first five epochs, gradually increasing to an initial learning rate of 0.4. Subsequently, the learning rate was decayed at epochs 60, 120, and 160, with a reduction factor of 0.1 at each step. Stochastic gradient descent (SGD) with Nesterov momentum (0.9) was used for optimization. In CaliMatch and OpenMatch, we applied two fixed thresholds ($\tau_1 = 0.5$ and $\tau_2 = 0.7$) used for OOD rejection and high quality of pseudo-labels. A weight decay coefficient of 0.0003 was used, incorporating L2 regularization on all model parameters. For data augmentation, we applied strong augmentation, including random color jitter, random grayscale, and random solarization, followed by an additional cutout operation. Weak augmentation was implemented using random horizontal flipping. During training, input images were randomly cropped and rescaled to 192×192, while during evaluation, images were center cropped and rescaled to 224×224, following standard ImageNet evaluation practices.

*https://github.com/bogus215/SafeSSL-Calibration.

## S-2. Additional Results and Discussions

**Calibration in Classification.** Table S-3 presents the average ECE and standard deviation obtained from five runs on four datasets. All SSL methods exhibited enhanced calibration in comparison to supervised learning across the four testing datasets. This improvement stems from the enhanced accuracy achieved through SSL methods, which reduced the discrepancy between relatively high confidence and actual accuracy. However, most comparison SSL methods failed to surpass the calibration improvements achieved by CaliMatch, especially on SVHN, CIFAR-10, and CIFAR-100 datasets. This highlights CaliMatch's effectiveness in selecting high-accuracy data with high confidence, a valuable trait in thresholding-based SSL. In the case of MTC, it also showed improved calibration that was comparable to our method in some cases. This can be attributed to the smoothing-based calibration method, mixup, used in MixMatch for MTC, which enhanced the calibration of deep CNN. However, it did not consider the selection of unlabeled samples using its well-calibrated confidence despite its potential in SSL.

Table S-1. Label distribution and number of data on five benchmark datasets under safe SSL setup.

| Dataset | Training labeled dataset | | | Training unlabeled dataset | | |
|---|---|---|---|---|---|---|
| | Label distribution | Number of data for each class | Total number of data | Label distribution | Number of data for each class | Total number of data |
| SVHN | 2,3,4,5,6,7 | 50 | 300 | 0-9 | 2,000 | 20,000 |
| CIFAR-10 | 2,3,4,5,6,7 | 400 | 2,400 | 0-9 | 2,000 | 20,000 |
| CIFAR-100 | 0-49 | 100 | 5,000 | 0-99 | 200 | 20,000 |
| TinyImageNet | 0-99 | 100 | 10,000 | 0-199 | 200 | 40,000 |
| ImageNet | 0-499 | 150 | 75,000 | 0-999 | 500 | 500,000 |

**Unseen-label Data Detection.** Table S-4 presents the average F1 and standard deviation of safe SSL methods to evaluate the unseen-label detection performance across five repeated runs with different random seeds. In the case of FixMatch and ADELLO, we defined their unseen-label score by subtracting confidence, which is the maximum probability value in multiclass classification, from one. Among the eight SSL methods, OpenMatch and CaliMatch showed the best or second-best performance across all datasets. MTC also showed satisfactory performance on SVHN and CIFAR-10, whose number of classes is relatively small when compared to CIFAR-100 and TinyImageNet. The F1 scores of MTC on CIFAR-100 and TinyImageNet indicate that MTC's OOD detector made incorrect decisions by assigning all testing samples to classes, which are in the label distribution of the labeled dataset. This suggests that training the OOD detector on datasets with a large number of similar classes can be unstable and may fail to distinguish between unknown classes and similarly known classes. When we implemented IOMatch using its official code in our experimental settings, we empirically observed that the projection head it utilized was sensitive to the performance of its OOD detector, thus negatively affecting unseen-label detection despite our efforts to find better hyperparameter settings for IOMatch.

**Learning Curves.** To demonstrate our CaliMatch's calibration performance in terms of safe SSL, we also present the learning curves of all SSL methods on SVHN, CIFAR-10, and TinyImageNet in Figure S-1. In Figure S-1, we can have a similar discussion with Figure 1 of the main paper by demonstrating CaliMatch's superiority and robustness across all datasets. To highlight the SSL methods with the best or second-best performance in each plot of Figure S-1, we sometimes do not show the results of other SSL methods in detail if their differences compared to the best or second-best methods are significant. In the case of MTC on CIFAR-10, it exhibited instability in learning, failing to sustain long-term training. The training was interrupted around the 80th epoch by a gradient exploding in its OOD detector.

**Sensitivity Analysis.** Table S-5 summarizes the results of sensitivity analysis on two hyperparameters ($\lambda_{\mathrm{O}}$ and $\lambda_{\mathrm{OCal}}$) in CaliMatch. We observed that the performance of safe SSL depends on the choice of $\lambda_{\mathrm{O}}$ and $\lambda_{\mathrm{OCal}}$, but no significant failures occurred within the range of hyperparameters considered in this analysis. This suggests that, while a naive choice of hyperparameters may not yield optimal performance, our method is robust enough to perform reliably in most cases. Furthermore, we confirmed that, across all hyperparameter combinations, CaliMatch consistently outperformed other safe SSL methods, such as OpenMatch and IOMatch, in the multiclass classification task.

**Safe SSL with Improved Calibration.** Model calibration aims to align a model's confidence scores with its actual accuracy, particularly when the model exhibits overconfidence or underconfidence. Effective calibration requires accurately assessing the discrepancy between the model's predicted confidence and its true accuracy, allowing the appropriate level of calibration to be applied. However, existing smoothing-based calibration methods, such as label smoothing and mixup, typically rely on fine-grained grid search to determine the optimal smoothing level but still fail to generalize well across different data distributions. Liu et al. [6] introduced a constrained-optimization approach to label smoothing and proposed margin-based LS (MbLS) for improved calibration. MbLS relaxes the equality constraint used in standard label smoothing by enforcing a more flexible inequality constraint, leading to better calibration. However, MbLS applies a fixed smoothing intensity to all instances, failing to adapt to variations across different data regions. This limitation can result in suboptimal calibration because different data samples require different levels of calibration. Similarly, Noh et al. [7] proposed RankMixup, which ranks mixup-augmented images and raw images based on their relative difficulty. Their method assumes that a model's confidence in mixup-generated samples should be lower than in raw images, helping to improve calibration. However, mixup-based calibration heavily depends on a hyperparameter controlling the degree of interpolation, making it sensitive to tuning and

Table S-2. Implementation details and hyperparameters on CIFAR-10, CIFAR-100, SVHN, and TinyImageNet.

| | |
|---|---|
| Shared | |
| Training iterations (epochs) | 500,000 (100) |
| Iteration period of validation | 5,000 |
| Learning rate | 0.003 |
| Learning rate decay factor | 0.2 |
| Learning rate decay at iteration | 400,000 |
| Optimizer | Adam |
| CNN backbone network | Wide ResNet 28-2 [14] |
| Batch size for labeled and unlabeled data on SSL | 50 |
| Supervised learning | |
| Batch size for labeled data | 100 |
| MTC | |
| Parameters ($\alpha$,$\beta$) for the Beta distribution for mixup | 0.75 |
| Temperature parameter for sharpening in MixMatch | 0.5 |
| Coefficient for mean squared error loss on unlabeled data | 75 |
| OOD detector | Single-layer perceptron |
| Pretraining iterations of OOD detector for stability | 50,000 |
| Finetuning iterations of OOD detector and MixMatch | 450,000 |
| FixMatch | |
| Threshold for selecting unlabeled training data | 0.95 |
| Coefficient for cross-entropy loss on unlabeled data | 1 |
| OpenMatch | |
| OOD detector | Single-layer perceptron |
| Threshold for FixMatch | 0.95 |
| Threshold for selecting seen-label data | 0.5 |
| Coefficient for FixMatch's loss on unlabeled data | 1 |
| Coefficient for entropy minimization of OOD detector | 0.1 |
| Coefficient ($\lambda_{\text{S}}$) for soft open-set consistency regularization loss | 0.5 |
| Warm-up training iterations for FixMatch | 25,000 |
| SafeStudent | |
| Temperature parameter for calculating energy discrepancy | $\{1,1.5\}$ |
| Pretraining iterations for teacher model | 100,000 |
| Exponential moving average (EMA) factor | 0.996 |
| Iteration period for the teacher model with EMA update | 50,000 |
| Coefficient for confirmation bias elimination loss | 1 |
| Coefficient for unseen-class label distribution learning loss | 0.01 |
| IOMatch | |
| OOD detector | Single-layer perceptron |
| Open-set classifier | Single-layer perceptron |
| Projection head | Three-layer perceptron |
| Threshold for FixMatch | 0.95 |
| Thresholds for selecting seen-label data | 0.5 |
| Coefficients for FixMatch | 1 |
| Coefficients for multi-binary and open-set classifiers | 1 |
| SCOMatch | |
| Size of OOD memory queue | max$\{8\times$number of classes,256$\}$ |
| Initial value for OOD detection and FixMatch | 0.95 |
| Positive and negative head classifier | Single-layer perceptron |
| ADELLO | |
| Exponential moving average decay | 0.999 |
| Minimum of progressive alpha | 0.1 |
| Value of progressive K | 2 |
| Threshold for FixMatch | 0.95 |
| **CaliMatch** | |
| OOD detector | Single-layer perceptron |
| Threshold ($\tau_2$) for FixMatch | 0.95 |
| Threshold ($\tau_1$) for selecting seen-label data | 0.5 |
| Coefficient ($\lambda_{\text{O}}$) for classification loss of OOD detector | $\{0.1,0.5,1\}$ |
| Coefficient ($\lambda_{\text{OCal}}$) for calibration loss of OOD detector | $\{0.0005,0.001,0.1\}$ |
| Coefficient ($\lambda_{\text{S}}$) for soft open-set consistency regularization loss | 0.5 |
| Epoch ($E_{\text{warm-up}}$) for warm-up stage in CaliMatch | 5 |
| Number ($M$) of bins for adaptive label smoothing | 30 |

Table S-3. Evaluation of multiclass classification using the averaged ECE and standard deviation (in parentheses) on four image benchmark datasets under two different existence rates ($\kappa\%$) of unseen-label data. The best results are in **bold**, and the second-best results are underlined.

| Dataset | $\kappa$ | Method | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Supervised | MTC | FixMatch | OpenMatch | SafeStudent | IOMatch | SCOMatch | ADELLO | **CaliMatch** |
| SVHN | 30% | 0.203 (0.008) | 0.054 (0.009) | 0.029 (0.002) | 0.021 (0.002) | 0.011 (0.010) | 0.025 (0.005) | 0.024 (0.006) | 0.027 (0.008) | **0.003 (0.002)** |
| | 60% | | 0.058 (0.008) | 0.036 (0.002) | 0.026 (0.002) | 0.016 (0.010) | 0.032 (0.002) | 0.033 (0.004) | 0.038 (0.008) | **0.006 (0.002)** |
| CIFAR-10 | 30% | 0.121 (0.011) | 0.085 (0.057) | 0.097 (0.004) | 0.096 (0.003) | 0.107 (0.010) | 0.083 (0.004) | 0.076 (0.006) | 0.091 (0.007) | **0.031 (0.004)** |
| | 60% | | 0.062 (0.008) | 0.116 (0.007) | 0.115 (0.007) | 0.123 (0.005) | 0.107 (0.009) | 0.109 (0.004) | 0.106 (0.005) | **0.029 (0.003)** |
| CIFAR-100 | 30% | 0.340 (0.005) | 0.046 (0.012) | 0.234 (0.004) | 0.226 (0.007) | 0.161 (0.010) | 0.216 (0.009) | 0.264 (0.010) | 0.233 (0.006) | **0.025 (0.008)** |
| | 60% | | 0.083 (0.036) | 0.257 (0.012) | 0.256 (0.006) | 0.171 (0.010) | 0.233 (0.007) | 0.276 (0.010) | 0.260 (0.007) | **0.025 (0.007)** |
| TinyImageNet | 30% | 0.513 (0.009) | **0.120 (0.015)** | 0.385 (0.007) | 0.388 (0.006) | 0.323 (0.010) | 0.363 (0.008) | 0.406 (0.008) | 0.398 (0.006) | 0.189 (0.008) |
| | 60% | | **0.133 (0.020)** | 0.408 (0.006) | 0.413 (0.008) | 0.338 (0.010) | 0.393 (0.007) | 0.420 (0.009) | 0.412 (0.009) | 0.233 (0.006) |

Table S-4. Evaluation of unseen-label data detection using the averaged F1 and standard deviation (in parentheses) on four datasets. The best results are in **bold**, and the second-best results are underlined.

| Dataset | Method | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | MTC | FixMatch | OpenMatch | SafeStudent | IOMatch | SCOMatch | ADELLO | CaliMatch |
| SVHN | 0.701 (0.068) | 0.200 (0.023) | 0.858 (0.009) | 0.676 (0.003) | 0.118 (0.037) | 0.867 (0.010) | 0.188 (0.024) | **0.889 (0.028)** |
| CIFAR-10 | 0.700 (0.029) | 0.140 (0.021) | 0.881 (0.013) | 0.695 (0.003) | 0.156 (0.063) | 0.503 (0.008) | 0.175 (0.033) | **0.883 (0.003)** |
| CIFAR-100 | 0.001 (0.000) | 0.486 (0.026) | **0.696 (0.002)** | 0.691 (0.002) | 0.419 (0.013) | 0.385 (0.016) | 0.442 (0.031) | 0.687 (0.006) |
| TinyImageNet | 0.001 (0.000) | 0.581 (0.008) | 0.688 (0.002) | 0.682 (0.004) | 0.627 (0.039) | 0.372 (0.008) | 0.570 (0.009) | **0.691 0.001** |

Table S-5. Sensitivity analysis of $\lambda_{\text{O}}$ and $\lambda_{\text{OCal}}$ for CaliMatch on CIFAR-10 with $\kappa = 60\%$.

| Coefficient | | Multiclass classification | | Unseen-label detection | |
|---|---|---|---|---|---|
| $\lambda_{\text{O}}$ | $\lambda_{\text{OCal}}$ | Accuracy | ECE | F1 | ECE |
| 0.1 | 0.1 | 87.62 (0.36) | 0.029 (0.003) | 0.883 (0.003) | 0.064 (0.005) |
| 0.1 | 0.05 | 87.94 (0.57) | 0.037 (0.010) | 0.881 (0.010) | 0.066 (0.013) |
| 0.1 | 0.01 | 87.74 (0.57) | 0.041 (0.012) | 0.873 (0.006) | 0.044 (0.006) |
| 0.5 | 0.1 | 87.51 (0.33) | 0.037 (0.013) | 0.875 (0.004) | 0.074 (0.014) |
| 1 | 0.1 | 86.86 (0.22) | 0.039 (0.011) | 0.872 (0.006) | 0.062 (0.011) |

potentially unreliable across diverse datasets.

In contrast, our calibration approach introduces a dynamic, adaptive mechanism that estimates the model's current accuracy at each training epoch using a labeled validation set. This enables the model to determine the appropriate level of label smoothing without requiring manual hyperparameter tuning. We align the model's confidence distribution with the estimated accuracy through adaptively smoothed labels with $T_M$ and $T_O$. The two learnable parameters optimize themselves to stabilize the calibration

Table S-6. Evaluation of multiclass classification and unseen-label detection using OpenMatch with calibration methods, as well as CaliMatch, on CIFAR-10 with $\kappa$ set to 60%. The best results are in **bold**, and the second-best results are underlined.

| Method | Calibration | | Multiclass classification | | Unseen-label detection | |
|---|---|---|---|---|---|---|
| | Multiclass classifier | OOD detector | Accuracy | ECE | F1 | ECE |
| OpenMatch | ✗ | ✗ | 86.19 (0.74) | 0.115 (0.007) | 0.881 (0.013) | 0.126 (0.006) |
| OpenMatch with label smoothing | ✓ | ✗ | 86.84 (0.40) | 0.082 (0.004) | 0.854 (0.008) | 0.121 (0.008) |
| | ✓ | ✓ | 83.49 (0.86) | 0.074 (0.017) | **0.887** **(0.015)** | 0.070 (0.014) |
| OpenMatch with mixup | ✓ | ✓ | 86.57 (0.36) | 0.060 (0.025) | 0.878 (0.024) | 0.065 (0.025) |
| OpenMatch with MbLS | ✓ | ✗ | 86.60 (0.51) | 0.092 (0.007) | 0.866 (0.011) | 0.108 (0.010) |
| OpenMatch with RankMixup | ✓ | ✗ | 87.13 (0.54) | 0.095 (0.008) | 0.829 (0.013) | 0.147 (0.014) |
| **CaliMatch** | ✓ | ✓ | **87.62** **(0.36)** | **0.029** **(0.003)** | 0.883 (0.003) | **0.064** **(0.005)** |

process as the models learn the adaptively smoothed labels. This stable characteristic would be a valuable factor when it is applied to other frameworks. To support our claims, we performed additional experiments on CIFAR-10 with the 60% mismatch ratio to compare our approach and other calibration methods. We applied various methods to improve calibration performance to OpenMatch. The results are summarized in Table S-6. We observed that all methods improved calibration performances compared to OpenMatch, but the improvement from our adaptive smoothing-based calibration was the best and eventually improved safe SSL. When the classic label smoothing was applied to both the multiclass classifier and OOD detector of OpenMatch, the OvR binary classifiers in the OOD detector exhibited instability because of gradient explosion, failing to sustain SSL training. This result highlights the importance of our learnable parameter $T_M$ in OOD calibration.

**Computational Complexity Analysis.** To compare the computational complexities of the training methods, we calculate the number of floating point operations (FLOPs) required for one iteration of each approach. Note that the experimental setting for this analysis is on CIFAR-10 with $\kappa$ set to 60%. As shown in Figure S-2, the supervised learning baseline requires 42.89 giga FLOPs (GFLOPs, where 1 GFLOP = $10^9$ FLOPs). Among all the SSL methods, FixMatch, IOMatch, and SafeStudent have moderate computational costs of approximately 64.34 GFLOPs, while CaliMatch and OpenMatch exhibit slightly higher complexity at 85.79 GFLOPs. This is because the two methods use soft consistency regularization for OOD detectors based on two weakly augmented unlabeled images, demonstrating better OOD detection performance compared to the other SSL methods. Although CaliMatch incorporates additional techniques such as label smoothing and logit scaling, their computational impact is negligible, leading to no significant difference in FLOPs between CaliMatch and OpenMatch. In contrast, SCOMatch incurs the highest computational cost at 214.41 GFLOPs, significantly surpassing the other SSL methods. This is because SCOMatch learns from not only existing labeled and unlabeled data, but also new additional data labeled as OOD samples from their proposed OOD memory queue, resulting in increased computational overhead.

**Thresholding in Safe SSL.** CaliMatch and OpenMatch have two threshold values $\tau_1$ and $\tau_2$ for safe SSL. Specifically, $\tau_1$ and $\tau_2$ are used to implement OOD rejection and FixMatch, respectively. On CIFAR-10 with $\kappa$ set to 60%, we investigated how the classification performance of CaliMatch and OpenMatch varies as the two threshold values change. As shown in Table S-7, although CaliMatch and OpenMatch exhibit some accuracy fluctuations on various $\tau_1$ and $\tau_2$ values, CaliMatch consistently achieves higher accuracy across all settings, particularly at higher thresholds.

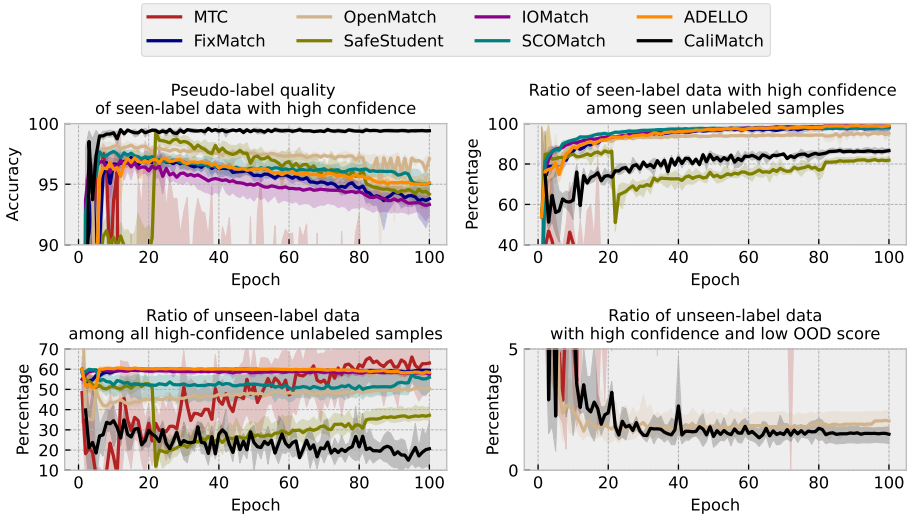
Table S-7. Performance variations in CaliMatch and OpenMatch on CIFAR-10 with $\kappa$ set to 60%. CaliMatch's results are in **bold**, and OpenMatch's results are in (parentheses).

| Threshold | | $\tau_2$ | | |
|---|---|---|---|---|
| | | 0.93 | 0.95 | 0.97 |
| $\tau_1$ | 0.5 | **87.36** (86.13) | **87.62** (86.22) | **87.90** (86.12) |
| | 0.6 | **87.63** (85.80) | **87.62** (86.29) | **87.84** (86.19) |
| | 0.7 | **87.25** (85.56) | **88.13** (86.22) | **87.85** (86.54) |
| | 0.8 | **87.25** (86.27) | **87.72** (86.37) | **87.95** (85.92) |

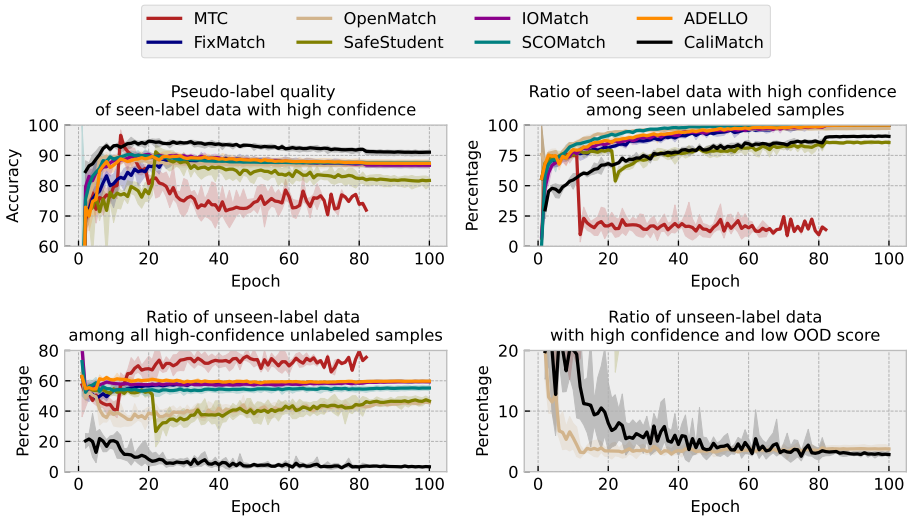
**Backbone Network Variation.** To further evaluate the performance of CaliMatch with two additional CNN backbone networks (DenseNet-121 [4] and ResNet-50), we conducted experiments on CIFAR-10 with $\kappa$ set to 60%. All methods (CaliMatch, OpenMatch, and the supervised baseline) shared the identical hyperparameter configuration as those used in the experiments conducted for Table 1 of the main paper. As shown in Table S-8, CaliMatch consistently outperforms both OpenMatch and the supervised approach in terms of accuracy, F1, and ECE across both backbone networks, demonstrating its robustness in the presence of unlabeled OOD samples during SSL. Notably, OpenMatch on DenseNet-121 failed to surpass the supervised baseline, highlighting its inability to mitigate the detrimental effects of unlabeled OOD samples during SSL. These results underscore CaliMatch's effectiveness and generality in handling safe SSL tasks across diverse backbone architectures.

Table S-8. Evaluation of multiclass classification and OOD detection of the baseline, OpenMatch, and CaliMatch with two popular CNNs on CIFAR-10 with $\kappa$ set to 60%. The best results are highlighted in **bold**. (ACC: Accuracy)

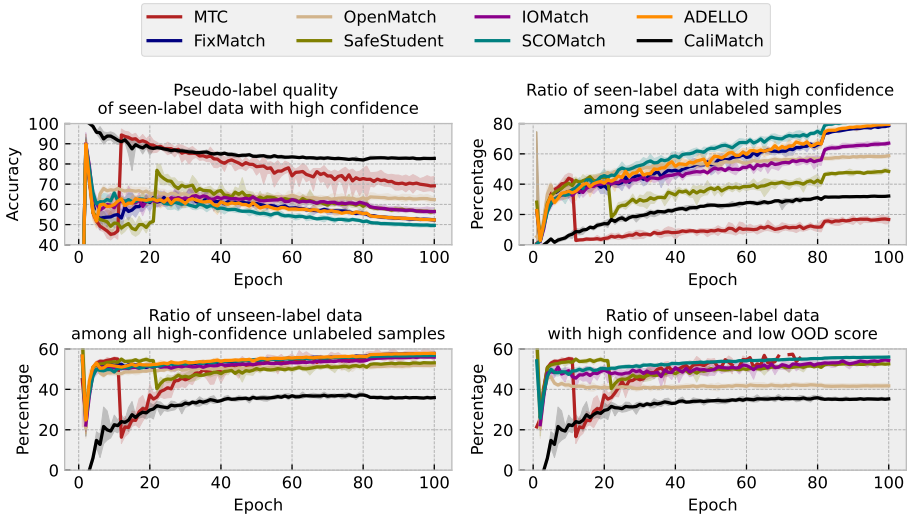
| Backbone | Method | Classification | | OOD detection | |
|---|---|---|---|---|---|
| | | ACC | ECE | F1 | ECE |
| ResNet-50 | Baseline | 56.41 | 0.409 | - | - |
| | OpenMatch | 58.72 | 0.307 | 0.740 | 0.131 |
| | **CaliMatch** | **72.73** | **0.123** | **0.784** | **0.120** |
| DenseNet-121 | Baseline | 75.20 | 0.226 | - | - |
| | OpenMatch | 71.60 | 0.224 | 0.782 | 0.189 |
| | **CaliMatch** | **80.81** | **0.070** | **0.831** | **0.096** |




(a) SVHN




(b) CIFAR-10




(c) TinyImageNet

Figure S-1. Learning curves averaged over five runs on SVHN, CIFAR-10, and TinyImageNet for CaliMatch and other SSL methods. The shaded region indicates standard deviations calculated from five runs.

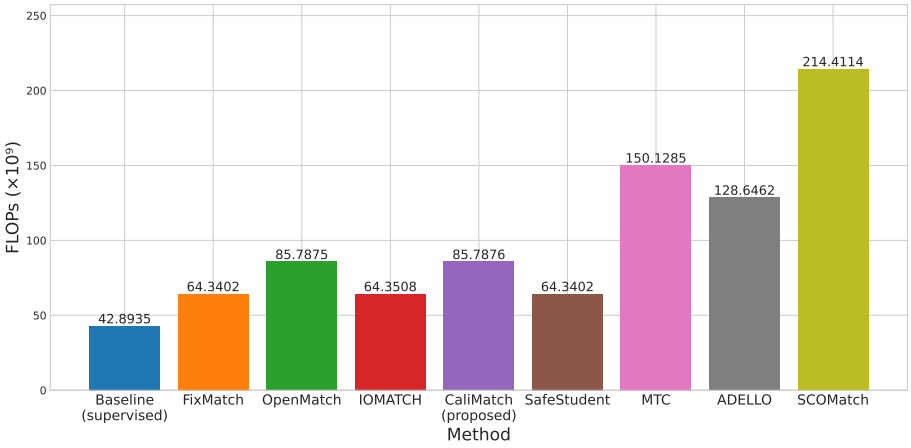



Figure S-2. Comparison of computational cost, measured in $10^9$ FLOPs, for SSL methods and a supervised baseline.

## S-3. Theoretical Justification

### S-3.1. Setups.

Let $D_u^a = \{(x_i^u, y_i^u) \in \mathcal{X} \times \tilde{\mathcal{Y}} : i = 1, \cdots, n_u\}$ be a set of unlabeled instances, where $\tilde{\mathcal{Y}}$ represents the set of all possible labels including the set of known labels $\mathcal{Y}$. We then consider $D_u^s = \{(x_i^u, y_i^u) \in D_u^a : y_i^u \in \mathcal{Y}\}$, which contains only in-distribution (ID) samples. By successfully applying an OOD detection method on $D_u^a$, we may obtain $D_u^s$. Then we can calculate cross-entropy loss over $D_u^s$ with strong augmentations $\mathcal{T}_s$ for classification tasks:

$$\mathcal{L}_{\text{CE}}(D_u^s; \mathcal{T}_s) = -\sum_{i=1}^{|D_u^s|} \sum_{k=1}^{K} y_{ik}^u \log p_k(\mathcal{T}_s(x_i^u)). \quad \text{(S-1)}$$

Training a classifier by minimizing $\mathcal{L}_{\text{CE}}(D_u^s; \mathcal{T}_s)$ as well as the cross-entropy loss on the labeled training dataset will provide further improvement compared to the case when we only consider cross-entropy loss on the labeled training dataset. However, it is challenging to identify the "ideal" dataset $D_u^s$ under safe SSL scenarios.

The safe SSL methods try to address this problem by having an OOD detector that identifies and discards instances from unseen classes within the unlabeled dataset $D_u$. They approximate the true labels $y_i^u$ for the remaining unlabeled instances in $D_u$ through pseudo-labeling techniques, which assign pseudo labels to unlabeled instances based on the model's predictions. In practice, these methods construct a surrogate loss function, $\mathcal{L}_{\text{Fix}}(B_u^t; \mathcal{T}_w, \mathcal{T}_s)$ (as seen in CaliMatch), which is designed to approximate the ideal loss $\mathcal{L}_{\text{CE}}(D_u^s; \mathcal{T}_s)$. One key consideration for the threshold-based safe SSL methods is that some unlabeled ID samples, despite having correct pseudo-labels, may be rejected from the surrogate loss computation because of low confidence. Here, we provide a theoretical analysis demonstrating how improving the calibration for both classification and OOD detection in safe SSL facilitates the alignment of gradients between the surrogate loss (provided by safe SSL) and the ideal loss (Equation (S-1)), computed on the subset of samples that satisfy the thresholds. This alignment is crucial because, under stochastic gradient descent, surrogate gradients that closely approximate the ideal gradients can lead to comparable optimization outcomes [1]. In other words, approximating the gradient of the ideal loss is sufficient to achieve a training effect similar to that obtained if the ideal loss were used directly.

### S-3.2. On the Importance of Calibration in Classification and OOD Detection.

The goal of safe SSL is to ensure the quality of pseudo-labeling and the accuracy of OOD rejection, particularly for samples that meet the confidence threshold and OOD rejection threshold. The following lemma establishes that improving model calibration reduces the probability of incorrect pseudo-labeling or the inclusion of OOD samples in the training set $B_u^t$.

**Lemma 1.** *Assume the model is well-calibrated in the sense that for any confidence level $s \in [0, 1]$, the empirical accuracy of samples with predicted confidence in an interval $[s - \delta, s + \delta]$ is approximately $s$ (with an error at most $\eta$ for sufficiently small $\delta > 0$). Here, we define $\varepsilon$ as follows:*

$$\varepsilon = P\left((x_i^u \in B_u^t) \wedge \left((y_i^u \notin \mathcal{Y}) \vee (y_i^u \neq \hat{y}_i^u)\right)\right),$$

*which represents the probability that a sample $x_i^u$ is either OOD or incorrectly pseudo-labeled. If the thresholds satisfy $\min\{\tau_1, \tau_2\} \geq 1-\eta$, then with probability at least $1-\varepsilon$, any $x_i^u$ from $B_u^t$ belongs to an ID class and is assigned the correct pseudo label:*

$$P\Big((x_i^u \in B_u^t) \Rightarrow (y_i^u \in \mathcal{Y}) \wedge (y_i^u = \hat{y}_i^u)\Big) \geq 1 - \varepsilon.$$

*Moreover, as model calibration improves (i.e., as calibration error $\eta$ decreases), the probability of incorrect selection $\varepsilon$ decreases.*

This lemma implies that with better calibration, the number of incorrectly pseudo-labeled or OOD samples in the training set is minimized, ensuring that the threshold-based selection process primarily retains correctly pseudo-labeled ID samples.

*Proof.* A well-calibrated model satisfies that for any interval $A \subset [0, 1]$ of predicted confidence scores, the empirical accuracy on $A$ is approximately equal to the mean confidence over $A$. Formally, if $A = \{x \mid s(x) \in [s - \delta, s + \delta]\}$, then, $\Big|P\big(y = \hat{y} \mid x \in A\big) - \mathbb{E}[s(x) \mid x \in A]\Big| \leq \eta$. For samples $x_i^u$ in $B_u^t$, the empirical accuracy of pseudo-label is at least $\tau_2 - \eta$. In a similar way, by the calibration assumption, the empirical OOD detection accuracy for selected samples is at least $\tau_1 - \eta$. Hence, the probability that a sample $x_i^u$ is either misclassified or OOD is bounded by

$$\varepsilon \leq 1 - \min\{\tau_1, \tau_2\} + \eta.$$

Thus, when the thresholds $\tau_1$ and $\tau_2$ are sufficiently high and the model is well-calibrated (i.e., $\eta$ is small), the probability of incorrect selection $\varepsilon$ is minimized. □

Next, we present a theorem that establishes how improved calibration facilitates the alignment of gradients between the surrogate and ideal loss functions. This alignment is crucial in ensuring that safe SSL optimization behaves similarly to the supervised learning.

**Theorem 1.** *Let $\mathcal{L}_{Fix}(B_u^t; \mathcal{T}_w, \mathcal{T}_s)$ be the FixMatch-based loss on $B_u^t$, and let $\mathcal{L}_{CE}(B_u^s; \mathcal{T}_s)$ be the ideal cross-entropy loss computed on all ID unlabeled data $B_u^s$ sampled from $D_u^s$ with true labels. Define $B_u^t \cap B_u^s$ as the subset of ID samples in $B_u^t$, i.e., the correctly classified ID samples that meet the thresholds. Then, under the assumption that $\varepsilon$ is sufficiently small, the gradient difference satisfies:*

$$\left\|\nabla_\theta \mathcal{L}_{Fix}(B_u^t; \mathcal{T}_w, \mathcal{T}_s) - \nabla_\theta \mathcal{L}_{CE}(B_u^t \cap B_u^s; \mathcal{T}_s)\right\| \leq C\varepsilon |B_u^t|,$$

*where $C$ is a positive constant.*

This theorem asserts that as the calibration error decreases, the gradients of the surrogate loss function become increasingly similar to those of the ideal loss, leading to more stable and reliable optimization in safe SSL settings.

*Proof.* Let $f_\theta(\mathcal{T}_s(x))$ produce logits $z_i = [z_{i,1}, \ldots, z_{i,K}]$ for $K$ classes, and denote $p_i = \mathrm{softmax}(z_i)$ as the predicted probabilities. The ideal cross-entropy loss for an ID sample $(x_i^u, y_i^u) \in B_u^s$ is given by:

$$\ell_{\text{CE}}(z_i, y_i^u) = -\log p_{i,y_i^u}.$$

For a sample $x_i^u \in B_u^t$ with pseudo-label $\hat{y}_i^u$, the surrogate loss is:

$$\ell_{\text{CE}}(z_i, \hat{y}_i^u) = -\log p_{i,\hat{y}_i^u}.$$

**Step 1: Gradient Expression at Sample Level** The gradient of the loss with respect to the logits is given by:

$$\frac{\partial \ell_{\text{CE}}(z_i, y)}{\partial z_{i,k}} = p_{i,k} - \mathbb{I}(y = k).$$

For correctly pseudo-labeled samples (i.e., $\hat{y}_i^u = y_i^u$), we have:

$$\frac{\partial \ell_{\text{CE}}(z_i, \hat{y}_i^u)}{\partial z_{i,k}} = \frac{\partial \ell_{\text{CE}}(z_i, y_i^u)}{\partial z_{i,k}},$$

meaning their gradient contributions to the surrogate and ideal losses are identical. For incorrectly pseudo-labeled samples ($\hat{y}_i^u \neq y_i^u$), the gradient error per sample is:

$$\Delta g_i = \frac{\partial \ell_{\text{CE}}(z_i, \hat{y}_i^u)}{\partial z_{i,k}} - \frac{\partial \ell_{\text{CE}}(z_i, y_i^u)}{\partial z_{i,k}}.$$

Since the difference between any two softmax gradients is bounded, there exists a constant $B$ such that:

$$\|\Delta g_i\| \leq B.$$

**Step 2: Batch-level Gradient Analysis** Now, we analyze the batch-level gradient over $B_u^t$ and compare it to the ideal batch gradient over $B_u^t \cap B_u^s$. The surrogate gradient over a batch $B_u^t$ is:

$$\nabla_\theta \mathcal{L}_{\text{Fix}}(B_u^t; \mathcal{T}_w, \mathcal{T}_s) = \sum_{x_i^u \in B_u^t} \nabla_\theta \ell_{\text{CE}}(z_i, \hat{y}_i^u).$$

The ideal gradient over a batch $B_u^t \cap B_u^s$ is:

$$\nabla_\theta \mathcal{L}_{\text{CE}}(B_u^t \cap B_u^s; \mathcal{T}_s) = \sum_{x_i^u \in B_u^t \cap B_u^s} \nabla_\theta \ell_{\text{CE}}(z_i, y_i^u).$$

By Lemma 1, with probability at least $1 - \varepsilon$, a sample in $B_u^t$ is correctly pseudo-labeled, meaning that for these samples, the gradient difference is zero. However, for at most $\varepsilon$-fraction of the samples, the gradient error per sample is at most $B$. Thus, the total batch-wise gradient difference satisfies:

$$\left\|\nabla_\theta \mathcal{L}_{\text{Fix}}(B_u^t) - \nabla_\theta \mathcal{L}_{\text{CE}}(B_u^t \cap B_u^s)\right\| \leq B\varepsilon |B_u^t|.$$

**Step 3: Extending to Model Parameters** Using the chain rule, since the Jacobian of the model parameters $\frac{\partial z}{\partial \theta}$ has a bounded norm $L$, the parameter-wise gradient difference satisfies:

$$\left\|\nabla_\theta \mathcal{L}_{\text{Fix}}(B_u^t) - \nabla_\theta \mathcal{L}_{\text{CE}}(B_u^t \cap B_u^s)\right\| \leq LB\varepsilon |B_u^t|.$$

Setting $C = LB$ completes the proof. □

## S-4. References


[1] Ahmad Ajalloeian and Sebastian U Stich. On the convergence of sgd with biased gradients. *arXiv preprint arXiv:2008.00051*, 2020. 9

[2] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020. 1

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[4] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017. 6

[5] Zekun Li, Lei Qi, Yinghuan Shi, and Yang Gao. Iomatch: Simplifying open-set semi-supervised learning with joint inliers and outliers utilization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15870–15879, 2023. 1

[6] Bingyuan Liu, Ismail Ben Ayed, Adrian Galdran, and Jose Dolz. The devil is in the margin: Margin-based label smoothing for network calibration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 80–88, 2022. 2

[7] Jongyoun Noh, Hyekang Park, Junghyup Lee, and Bumsub Ham. Rankmixup: Ranking-based mixup training for network calibration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1358–1368, 2023. 2

[8] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. *Advances in Neural Information Processing Systems*, 31, 2018. 1

[9] Kuniaki Saito, Donghyun Kim, and Kate Saenko. Openmatch: Open-set semi-supervised learning with open-set consistency regularization. *Advances in Neural Information Processing Systems*, 34:25956–25967, 2021. 1

[10] Emanuel Sanchez Aimar, Nathaniel Helgesen, Yonghao Xu, Marco Kuhlmann, and Michael Felsberg. Flexible distribution alignment: Towards long-tailed semi-supervised learning with proper calibration. In *European Conference on Computer Vision*, pages 307–327. Springer, 2024. 1

[11] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems*, 33:596–608, 2020. 1

[12] Zerun Wang, Liuyu Xiang, Lang Huang, Jiafeng Mao, Ling Xiao, and Toshihiko Yamasaki. Scomatch: Alleviating overtrusting in open-set semi-supervised learning. In *European Conference on Computer Vision*, pages 217–233. Springer, 2024. 1

[13] Qing Yu, Daiki Ikami, Go Irie, and Kiyoharu Aizawa. Multi-task curriculum framework for open-set semi-supervised learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pages 438–454. Springer, 2020. 1

[14] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *Proceedings of the British Machine Vision Conference*, pages 87.1–87.12, 2016. 3