OutlierGuard: Consistency-Guided Semi-supervised Learning in the Presence of Outliers

Abstract

Semi-supervised learning (SSL) is a powerful approach for enhancing a model's performance by utilizing unlabeled data. Traditional SSL methods, such as FixMatch, operate under the assumption that labeled and unlabeled data share the same label space. However, in real-world scenarios, unlabeled data may contain categories not present in the labeled set, known as outliers, which can significantly impact SSL algorithm performance.To address this challenge, we introduce a novel approach to Open-set Semi-Supervised Learning (OSSL) called OutlierGuard. The success of OSSL relies on learning representations of inliers while effectively rejecting outliers. OutlierGuard combines FixMatch with novelty detection using one-vs-all (OVA) classifiers. The OVA-classifier outputs confidence scores indicating whether a sample is an inlier, providing a threshold for outlier detection.An additional key contribution is the introduction of an open-set soft-consistency regularization loss, which enhances the smoothness of the OVA-classifier concerning input transformations. This regularization greatly improves outlier detection. OutlierGuard demonstrates state-of-the-art performance across three datasets and even surpasses a fully supervised model in detecting outliers not present in unlabeled data, as observed on the CIFAR10 dataset.

Keywords: OSSL, OVA-classifiers, outlier detection.

1 Introduction

Semi-supervised learning (SSL) is a powerful technique that enhances a model's performance by utilizing unlabeled data. It propagates class information from a small labeled set to a larger unlabeled set, improving recognition accuracy without additional annotation costs. However, SSL assumes identical label spaces for labeled and unlabeled data, which may not hold true in practice due to the presence of novel categories or outliers.

To address this, we introduce OutlierGuard, a framework for Open-set Semi-Supervised Learning (OSSL). Unlike existing SSL methods that struggle with OSSL, OutlierGuard utilizes a One-Vs-All (OVA) network to learn a threshold for distinguishing outliers from inliers in an unsupervised manner. Additionally, we propose a novel open-set soft-consistency loss to enhance outlier detection by improving the smoothness of the outlier detector function. This loss encourages consistency between logits obtained from differently transformed unlabeled inputs.

OutlierGuard achieves consistent improvements over baselines in various OSSL datasets and settings. For example, on CIFAR10 with 300 labeled examples, OutlierGuard achieves a 10.4% error rate compared to the previous state-of-the-art of 20.3%. Notably, it performs well in detecting outliers unseen in unlabeled data, outperforming a supervised model by 3.4% higher AUROC in experiments with 100 labels per class on CIFAR10.

In summary, our contributions include the introduction of a soft open-set consistency regularization (SOCR), the OutlierGuard framework combining OVA-classifier, SOCR, and FixMatch, and achieving a new state-of-the-art in correctly classifying inliers and detecting outliers, even when the outliers are unseen in unlabeled training data.

2 Related works

2.1 Semi-supervised Learning

Pseudo-labeling (PL) serves as a robust baseline in semi-supervised learning, and approaches like FixMatch or UDA, which integrate data augmentation and PL, demonstrate top performance on various benchmark datasets. However, even these powerful methods face challenges when confronted with noisy unlabeled data containing novel categories. Conversely, techniques based on soft consistency regularization aim to ensure a smooth decision boundary concerning stochastic transformations or models. The term ”soft” implies the absence of sharpening or pseudo-labeling on logits, utilizing soft targets to propagate training signals. This soft consistency regularization aligns with an approximated Jacobian regularization approach.

2.2 Open-set SSL

Open-set Semi-Supervised Learning (OSSL) methods encompass MTC, D3SL, and UASD. MTC adopts a strategy of updating network parameters and anomaly scores for unlabeled data alternately. It also minimizes the SSL loss (similar to MixMatch) for a subset of unlabeled data identified as inliers. D3SL, on the other hand, selectively utilizes unlabeled data to optimize SSL loss while also optimizing a function for selecting relevant unlabeled data. UASD generates soft targets for inliers (closed-set classifier) by averaging predictions from temporally ensembled networks.

2.3 Open-set Domain Adaptation(ODA)

Open-set Semi-Supervised Learning (OSSL) shares similarities with open-set domain adaptation in that both involve unlabeled data containing novel categories. However, a key distinction lies in the fact that unlabeled and labeled data in OSSL follow different data distributions. Another notable difference is the objective: OSSL aims to train a model from scratch, while domain adaptation tasks assume access to models pre-trained on ImageNet.

2.4 Novelty Detection

Novelty detection or open-set classification aims to identify outliers that were entirely unseen during training. Self-supervised learning methods like rotation prediction and contrastive learning have proven useful in distinguishing outliers from inliers. This task assumes the availability of plenty of labeled inliers in the training data. Padhy et al. employ OVA-classifiers for this purpose. Hendrycks et al. reveal that exposing a model to outlier data enables effective anomaly detection and training on out-of-distribution data in a supervised manner. However, obtaining such valuable out-of-distribution data is not always feasible. In Open-set Semi-Supervised Learning (OSSL), a model cannot be trained on numerous labeled samples and must instead detect which samples are inliers or outliers in unlabeled data. This makes OSSL more challenging and realistic in comparison.

3 Method

3.1 Overview

Our model has three components: (1) a shared feature extractor $F(\cdot)$, (2) an outlier detector consisting of K one-vs-all (OVA) sub-classifiers $D^j(\cdot)$, $j \in \{1, ..., K\}$, (3) a closed-set classifier C($\cdot$) which outputs a probability vector $\in R^K$ for K-class classification. At test time, the closed-set classifier is first applied to predict the K-way label, $\hat{y}$. If $D^{\hat{y}}$ predicts an inlier, then the output class is $\hat{y}$, otherwise, the output is ”outlier”. The main technical novelty comes from the choice of OVA-classifiers for outlier detection as well as training them with soft open-set consistency regularization. We first describe the training of the OVA-classifiers before describing the remaining details of our framework.

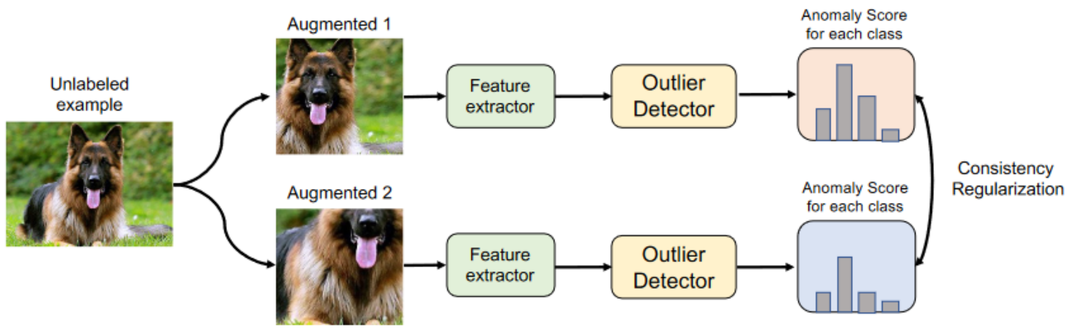


Figure 1. An illustration of our proposed open-set soft-consistency loss used to enhance outlier detection.

3.2 One-vs-All Outlier Detector

For open-set or universal domain adaptation, Saito et al. propose a method where OVA-classifiers are trained to detect outlier samples in an unlabeled target domain. The objective is to learn a boundary between inliers and outliers for each class. Each sub-classifier is trained to determine whether a sample is an inlier for the corresponding class. For instance, the sub-classifier for class j, denoted as D^j, outputs a 2-dimensional vector $z_b^j = D^j(F(x_b)) \in R^2$, where each dimension indicates the score of a sample being an inlier and outlier, respectively. The probabilities $p^j(t = 0|x_b)$ and $p^j(t = 1|x_b)$ represent the likelihood of x_b being the inlier and outlier for class j, computed by Softmax(z_b^j) (Note that $p^j(t = 0|x_b) + p^j(t = 1|x_b) = 1$).

The training of the sub-classifier Dj involves treating data from class j as positives and data from all other classes as negatives. The loss function for the outlier detector in a given batch $X := \{x_b, y_b\}_b^B = 1$ is defined as:

$L_{ova}(X) := \frac{1}{B}\sum_{b=1}^{B}(-\log(p^{y_b}(t=0|x_b)) - \min_{i\neq y_b}\log(p^i(t=1|x_b)))(1)$

Here, a hard-negative sub-classifier sampling technique is employed to effectively learn the threshold. The key is that each sub-classifier outputs a distance representing how far the input is from the corresponding class, making the classifiers effective in identifying unlabeled outliers.

The output of D^j is utilized to detect outliers based on the following criteria: If the probability of a sample being an outlier for the class highest-scored by $C(\cdot)$ is greater than the probability of being an inlier, the input is considered an outlier. Specifically, an unlabeled sample u_b is identified as an outlier if $(\hat{\{p\}}_{\{y\}}(t=0|u_b) < 0.5)$, where $(\hat{y} = \text{argmax}_j C(F(u_b)))$.

Open-set Entropy Minimization. For unlabeled samples, entropy minimization with respect to the OVA-classifiers, denoted as $L_{em}(U)$. The goal is to enhance the separation between inliers and outliers through the minimization of this loss for a given batch $U := \{u_b\}_{\mu}^{B}$:

$$L_{em}(U) = -\frac{1}{\mu B}\sum_{b=1}^{\mu B}\sum_{j=1}^{K} p^j(t=0|u_b)\log p^j(t=0|u_b) + p^j(t=1|u_b)\log p^j(t=1|u_b)(2)$$

The distinction between our approach and [32] lies in the fact that they utilize a model pre-trained on ImageNet, which extracts highly discriminative features. Consequently, inliers and outliers are well-separated even before fine-tuning. In contrast, our objective is to train a model to learn representations from scratch, where inliers and outliers are initially confused. Therefore, we require an objective to ensure their separation in addition to Equation 2.

3.3 Soft Open-set Consistency Regularization(SOCR)

To illustrate the distinction between SSL and OSSL, we present Fig. 2. In SSL, hard consistency losses like pseudo-labeling can effectively propagate label information from labeled to unlabeled samples. Through these losses, unlabeled samples receive training signals from neighboring labeled samples (Fig. 2(a)). On the contrary, in OSSL, as outliers are not assigned any labels, labeled samples are distant from outliers, rendering hard labeling unreliable and leading to potentially incorrect unsupervised training signals (Fig. 2(b)). The open-set entropy minimization described earlier can also be viewed as a form of hard consistency, as it is minimized when the predicted output class distribution is one-hot. To propagate useful training signals to unlabeled samples in OSSL, we propose smoothing the decision boundary of the outlier detector by minimizing the distance between its predictions on two augmentations of the same image (Fig. 2(c)). We employ unsharpened logits, maintaining a soft training signal to prevent the introduction of incorrect pseudo-labels. This loss is termed soft open-set consistency regularization (SOCR). With the smoothed outlier detector, the expectation is that the training signals from open-set entropy minimization will also become more accurate.

Specifically, soft open-set consistency regularization (SOCR) improves the smoothness of the outlier detector over data augmentation T, where, in our experiments, standard random cropping is used as T. It's important to note that standard cropping is also employed in computing Eq. 1 and 2, but we omit it for simplicity of notation.

To implement SOCR, we generate two different views of u_b, namely $T_1(u_b)$ and $T_2(u_b)$, where T_1 and T_2 are data augmentation functions stochastically sampled from T. Let $p^j(t|T_1(u_b))$ be the output logits of $T_1(u_b)$ from the j-th OVA-classifier. SOCR encourages the consistency of output logits over T to enhance smoothness by minimizing the following loss for a given batch $U := \{u_b\}_{b=1}^{\mu B}$:

$$L_{oc}(U,T) = \frac{1}{\mu B}\sum_{b=1}^{\mu B}\sum_{j=1}^{K}\sum_{t\in\{0,1\}} |p^j(t|T_1(u_b)) - p^j(t|T_2(u_b))|^2$$

This minimizes the distance between two logits to ensure smoothness concerning T. It's crucial to highlight that the regularization does not involve any sharpening of the output logits, keeping them soft. A notable distinction from previous soft consistency regularization methods is that we use the outlier detector to compute the regularization loss.

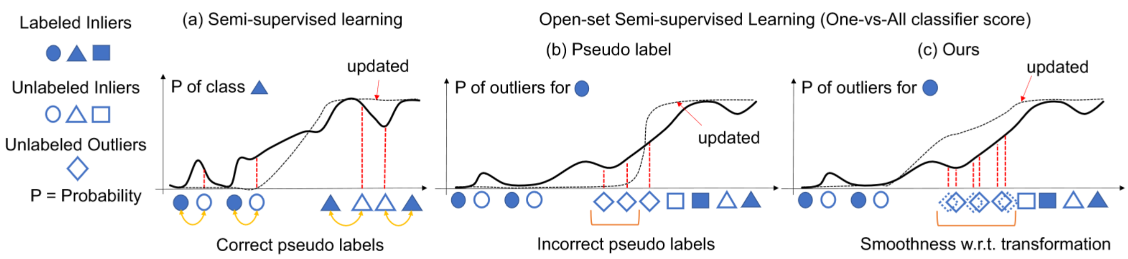


Figure 2. (a) SSL with pseudo-labels. Hard labels such as pseudo-labels are often used in SSL to propagate training signals from labeled samples to neighboring unlabeled ones. (b) OSSL with pseudo-labels. However, in OSSL, the outlier unlabeled samples do not have any labeled neighbors. Therefore, the pseudo-labels can be highly unreliable. (c) OSSL with soft consistency (ours). To address this, we propose to ensure smoothness with respect to data augmentation by minimizing a soft consistency loss. This separates the outliers from inliers and avoids using incorrect pseudo-labels.

4 Implementation details

4.1 Comparing with the released source codes

In the OutlierGuard implementation, the following points highlight potential improvements and features:

1. Outlier Detector Integration: The model incorporates an additional component, the 'outlier_detector', dedicated to detecting outliers in the feature space. The 'outlier_loss' function introduces a hypothetical outlier loss, emphasizing the importance of effectively learning representations for detecting outliers.

2. Unified Framework: The 'OutlierGuardModel' unifies one-vs-all classifiers and FixMatch, addressing the challenge of open-set semi-supervised learning (OSSL). The model architecture aims to handle both labeled and unlabeled data, emphasizing the importance of distinguishing known and unknown categories.

3. Training Logic: The training function ('train_outlierguard') demonstrates the joint optimization of classification and outlier detection tasks, reflecting the essence of OSSL. The utilization of the 'outlier_loss' alongside the classification loss emphasizes the model's ability to handle novel categories in unlabeled data.

4. Evaluation Metrics: The validation function ('validate_outlierguard') calculates both classification accuracy for known classes and AUROC for outlier detection. This dual evaluation strategy provides a comprehensive assessment of the model's performance on both traditional and open-set classification tasks.

5. Flexibility and Customization: - The code provides a foundation for further customization, allowing for the incorporation of advanced techniques, diverse architectures, and alternative outlier detection methods.

It's important to note that the actual implementation may require fine-tuning, detailed experimentation, and validation on specific datasets. The outlined code serves as a starting point, and its effectiveness would be further validated through empirical results and experiments.

4.2 Experimental environment setup

We assess the effectiveness of OutlierGuard on various Semi-Supervised Learning (SSL) image classification benchmarks, conducting experiments with different amounts of labeled data and varying numbers of known/unknown classes on CIFAR10/100 and ImageNet. We aim to cover diverse scenarios using these datasets. Hyperparameters are kept identical across experiments, with oc tuned on each dataset, and em set to 0.1. Additionally, fm is set to 0 before Efix epochs and then to 1 for all experiments, with Efix set to 10 in all cases. The hyperparameters for FixMatch, such as data augmentation, confidence threshold, are consistent for simplicity, tuned on a validation set without outliers. The complete set of hyperparameters is available in the appendix. Each experiment is conducted on a single 12-GB GPU, such as an NVIDIA TitanX.

As a baseline for Open-Set Semi-Supervised Learning (OSSL), we utilize MTC with the author's implementation. Additionally, we include the result of a model trained exclusively with labeled samples (Labeled Only). UASD and D3SL are excluded from baselines due to reported results underperforming a model trained only with labeled samples. The comparison with FixMatch aims to highlight the impact of OVA-classifiers trained with the soft consistency loss. Hyperparameters for both OutlierGuard and baselines are tuned by maximizing accuracy on the validation set, which doesn't contain outliers.

For evaluation, we assume the test set includes both known (inlier) and unknown (outlier) classes. Classification accuracy is used for known classes, while AUROC is employed to assess the separation into inliers and outliers, following the standard evaluation protocol of novelty detection. The anomaly score is computed using the outlier detector's score for OutlierGuard, Labeled Only, and FixMatch (where our outlier detector is added to the latter two models). Results are reported as averages over three runs with standard deviations.

4.3 Algorithm design

The entire OpenMatch algorithm is delineated in Alg. 1. The unsupervised losses (Eqs. 2, 3) employed for the outlier detector prove effective in identifying outliers within unlabeled data. However, these losses alone lack adequacy in correctly classifying unlabeled inliers. To address this, we propose the introduction of a semi-supervised learning loss specifically designed for unlabeled samples identified as inliers.

We opt for FixMatch as our choice, given its simplicity and robust performance in semi-supervised learning. FixMatch initially generates pseudo-labels by leveraging the model's predictions on weakly augmented unlabeled images. Subsequently, the model undergoes training to predict these pseudo-labels when presented with a strongly augmented version of the same image. The term "weak" denotes augmentations like simple random cropping, while "strong" implies more extensive data augmentations, such as RandAugment and CTAugment.

For labeled samples, we compute the standard cross-entropy loss to train C$(\cdot), denotedas \mathrm{L}_{cls}(X)$, for the closed-set output, $C(F(x_b))$. We aggregate $L_{sup}(X)$ as the sum of $L_{ova}(X)$ and $L_{cls}(X)$. Following E_{fix}

epochs of training with L_{sup}, L_{em}, L_{oc}, we initiate the selection of pseudo-inliers from unlabeled data at each subsequent epoch. Concretely, in Alg. 1, the process denoted as Select(ω, D_u) involves classifying unlabeled samples using the model's parameters w, encompassing the models (F(·), D(·), and C(·)), and the unlabeled data.

Subsequently, the FixMatch loss, denoted as Lfm(I), is exclusively computed for the pseudo-inliers to optimize the model, where I represents a batch of pseudo-inliers. The resulting comprehensive objective of OpenMatch is expressed as:

$$L_{\text{all}}(X, U, T, I) := L_{\text{sup}}(X) + \lambda_{\text{em}} L_{\text{em}}(X) + \lambda_{\text{oc}} L_{\text{oc}}(U, T) + \lambda_{\text{fm}} L_{\text{fm}}(I)$$

Here, λ_{em}, λ_{oc}, and λ_{fm} serve as control parameters, determining the trade-off for each specific objective.

4.4 Main contributions

The main contributions of this paper can be summarized as follows:

1. OpenSet Semi-Supervised Learning (OSSL): Introduction of a new framework, OpenSet Semi-Supervised Learning (OSSL), to address the presence of unknown categories in unlabeled data. This represents a more realistic scenario compared to traditional Semi-Supervised Learning (SSL).

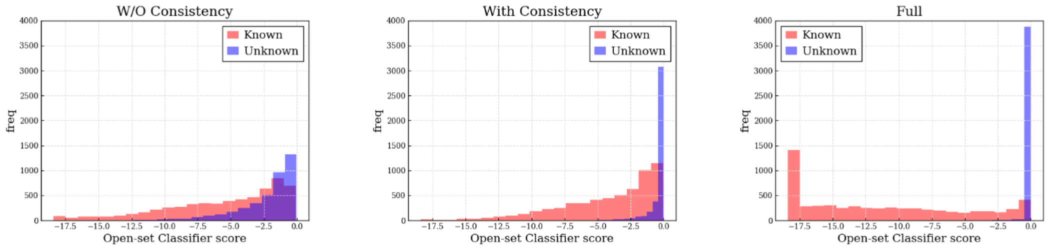
2. OutlierGuard Framework: Proposal of a novel framework called OutlierGuard designed to tackle the challenge of unknown categories between labeled and unlabeled data in OSSL. The framework combines One-Vs-All (OVA) classifiers, open-set soft-consistency regularization loss, and the FixMatch method.

3. One-Vs-All (OVA) Classifiers and Soft Consistency Loss: Introduction of One-Vs-All classifiers to learn a threshold for distinguishing inliers from outliers. - Introduction of an open-set soft-consistency regularization loss, enhancing the smoothness of the OVA classifier with respect to input transformations and significantly improving outlier detection.

4. Integration of FixMatch: Integration of the FixMatch method, a powerful semi-supervised learning approach. FixMatch generates pseudo-labels and optimizes the model through strong data augmentation, addressing the OSSL challenge of unknown categories in unlabeled inliers.

These contributions collectively form a comprehensive approach to OpenSet Semi-Supervised Learning, providing advancements in outlier detection, model regularization, and effective utilization of unlabeled data with unknown categories.

5 Results and analysis



(a) w/o FixMatch, SOCR. (b) w/o FixMatch (c) FULL (OpenMatch)

Figure 3. The histograms of the outlier detector's scores obtained with ablated models. Red: Inliers, Blue: Outliers. From left to right, a model without FixMatch and SOCR, a model without FixMatch, and a model with all objectives. These results show that SOCR ensures separation between inliers and outliers, and FixMatch added to SOCR can further enhance this separation.

Dataset	CIFAR10			CIFAR100		CIFAR100		ImageNet-30
No. of Known / Unknown	6 / 4			55 / 45		80 / 20		20 / 10
No. of labeled samples	50	100	400	50	100	50	100	10 %
Labeled Only	35.7±1.1	30.5±0.7	20.0±0.3	37.0±0.8	27.3±0.5	43.6±0.5	34.7±0.4	20.9±1.0
FixMatch [35]	43.2±1.2	29.8±0.6	16.3±0.5	35.4±0.7	27.3±0.8	41.2±0.7	34.1±0.4	12.9±0.4
MTC [44]	20.3±0.9	13.7±0.9	9.0±0.5	33.5±1.2	27.9±0.5	40.1±0.8	33.6±0.3	13.6±0.7
OpenMatch	**10.4**±0.9	**7.1**±0.5	**5.9**±0.5	**27.7**±0.4	**24.1**±0.6	**33.4**±0.2	**29.5**±0.3	**10.4**±1.0

Table 1. Tale 1:Error rates (%) with standard deviation for CIFAR10, CIFAR100 on 3 different folds. Lower is better. For ImageNet, we use the same fold and report averaged results of three runs. Note that the number of labeled samples per each class is shown in each column.

Dataset	CIFAR10			CIFAR100		CIFAR100		ImageNet-30
No. of Known / Unknown	6 / 4			55 / 45		80 / 20		20 / 10
No. of labeled samples	50	100	400	50	100	50	100	10 %
Labeled Only	63.9±0.5	64.7±0.5	76.8±0.4	76.6±0.9	79.9±0.9	70.3±0.5	73.9±0.9	80.3±1.0
FixMatch [35]	56.1±0.6	60.4±0.4	71.8±0.4	72.0±1.3	75.8±1.2	64.3±1.0	66.1±0.5	88.6±0.5
MTC [44]	96.6±0.6	98.2±0.3	98.9±0.1	81.2±3.4	80.7±4.6	79.4±2.5	73.2±3.5	93.8±0.8
OpenMatch	**99.3**±0.3	**99.7**±0.2	**99.3**±0.2	**87.0**±1.1	**86.5**±2.1	**86.2**±0.6	**86.8**±1.4	**96.4**±0.7

Table 2. AUROC of Table 1. Higher is better. Note that the number of labeled samples per each class is shown in each column.

Dataset	CIFAR10		CIFAR100		ImageNet-30
No. Known / Unknown	6 / 4		80 / 20		20 / 10
No. Labeled samples	50	400	50	100	10 %
without SOCR	60.5±2.8	75.8±0.8	70.4±0.1	73.2±0.2	81.3±0.4
with SOCR	**81.3**±2.9	**96.8**±0.6	**78.9**±0.1	**85.0**±0.8	**89.3**±0.3

Table 3. Ablation study of our soft consistency regularization (SOCR, Loc). We report AUROC scores (%). In this study, we do not apply FixMatch to pseudo-inliers to see the pure gain from SOCR.

		Unseen Out-liers				
Method	CIFAR10	SVHN	LSUN	ImageNet	CIFAR100	MEAN
Labeled Only	64.7±1.0	83.6±1.0	78.9±0.9	80.5±0.8	80.4±0.5	80.8±0.8
FixMatch [35]	60.4±0.4	79.9±1.0	67.7±2.0	76.9±1.1	71.3±1.1	73.9±1.3
MTC [44]	98.2±0.3	87.6±0.5	82.8±0.6	96.5±0.1	90.0±0.3	89.2±0.4
OpenMatch	**99.7±0.1**	**93.0**±**0.4**	**92.7±0.3**	**98.7±0.1**	**95.8±0.4**	**95.0±0.3**
Supervised	89.4±1.0	95.6±0.5	89.5±0.7	90.8±0.4	90.4±1.0	91.6±0.6

(a) Model trained on CIFAR10 (100 labeled data per class and unlabeled data.)

		Unseen Out-liers						
Method	ImageNet-30	LSUN	DTD	CUB	Flowers	Caltech	Dogs	MEAN
Labeled Only	80.3±0.5	85.9±1.4	75.4±1.0	77.9±0.8	69.0±1.5	78.7±0.8	84.8±1.0	78.6±1.1
FixMatch [35]	88.6±0.5	85.7±0.1	83.1±2.5	81.0±4.8	**81.9±1.1**	83.1±3.4	86.4±3.2	83.0±1.9
MTC [44]	93.8±0.8	78.0±1.0	59.5±1.5	72.2±0.9	76.4 ±2.1	80.9±0.9	78.0±0.8	74.2±1.2
OpenMatch	**96.3±0.7**	**89.9**±1.9	**84.4**±0.5	**87.7**±1.0	80.8±1.9	**87.7±0.9**	**92.1±0.4**	**87.1**±1.1
Supervised	92.8±0.8	94.4±0.5	92.7±0.4	91.5±0.9	88.2±1.0	89.9±0.5	92.3±0.8	91.3±0.7

(b) Model trained on ImageNet-30 (10 % of labeled data and unlabeled data).

Table 4. Evaluation of outlier detection on outliers unseen in unlabeled training data (AUROC). Higher is better. Supervised models use the same batch size, learning rate as OpenMatch, but are trained with fully labeled inliers.

6 Conclusion and future work

In this paper, we present a method for Open-Set Semi-Supervised Learning (OSSL), addressing the presence of samples from novel categories in unlabeled data. Our approach introduces a novel framework called OutlierGuard, which combines one-vs-all classifiers and FixMatch. The proposed objective, open-set soft consistency loss, proves effective in detecting outliers from unlabeled data, enabling FixMatch to perform well in the OSSL context. Additionally, OutlierGuard demonstrates the ability to detect outliers not seen in unlabeled data better than a supervised model. We anticipate that our OSSL framework will enhance the practicality of label-efficient techniques.

Limitations. OutlierGuard may encounter challenges in detecting outliers that closely resemble inliers, a limitation shared by various outlier detection methods. When outliers exhibit visual characteristics similar to inliers, distinguishing between them becomes challenging. A possible remedy is to incorporate self-supervised learning for unlabeled data.

References

References

[1] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. *Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. arXiv preprint arXiv:1911.09785*, 2019.

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. *Mixmatch: A holistic approach to semi-supervised learning. arXiv preprint arXiv:1905.02249*, 2019.

[3] Pau Panareda Busto and Juergen Gall. *Open set domain adaptation.* In *Int. Conf. Comput. Vis.*, 2017.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. *A simple framework for contrastive learning of visual representations.* In *Int. Conf. Mach Learn.*, pages 1597–1607. PMLR, 2020.

[5] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. *Big self-supervised models are strong semi-supervised learners. arXiv preprint arXiv:2006.10029*, 2020.

[6] Kuniaki Saito, Donghyun Kim, Kate Saenko *OpenMatch: Open-set Consistency Regularization for Semi-supervised Learning with Outliers arXiv:2105.14148*, 2021.