THESIS FOR THE DEGREE OF LICENTIATE OF ENGINEERING

Semi-supervised learning with self-supervision for closed and open sets

Erik Wallin

Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden, 2023

Semi-supervised learning with self-supervision for closed and open sets

Erik Wallin

This thesis has been prepared using LaTeX.

Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden
Phone: +46 (0)31 772 1000
www.chalmers.se

Printed by Chalmers Digitaltryck
Gothenburg, Sweden, July 2023

Abstract

Semi-supervised learning (SSL) is a learning framework that enables the use of unlabeled data with labeled data. These methods play a crucial role in reducing the burden of human labeling in training deep learning models. Many methods for SSL learn from unlabeled data through confidence-based pseudo-labeling. This technique involves assigning artificial labels to unlabeled data based on model predictions, given that these predictions exceed a confidence threshold. A drawback of this approach is that large parts of data may be ignored. This work proposes a self-supervised component for these frameworks to enable learning from all unlabeled data. The proposed self-supervision involves aligning feature predictions across weak and strong data augmentations for each unlabeled sample. We show that this approach, DoubleMatch, leads to improved training speed and accuracy on many benchmark datasets.

SSL is often studied in the closed-set scenario, where we assume that unlabeled data only contain classes present in the labeled data. More realistically, there is a risk that unlabeled data contain unseen classes, corrupted data, or outliers in other forms. This setting is referred to as open-set semi-supervised learning (OSSL). Many existing methods for OSSL use a procedure that involves selecting samples from unlabeled data that likely belong to the known classes, for inclusion in a traditional SSL objective. This work proposes an alternative approach, SeFOSS, that utilizes all unlabeled data through the inclusion of the self-supervised component proposed by DoubleMatch. Additionally, SeFOSS uses an energy-based method for classifying data as in-distribution (ID) or out-of-distribution (OOD). Experimental evaluation shows that SeFOSS achieves strong results for both closed-set accuracy and OOD detection in many open-set scenarios. Additionally, our results indicate that traditional methods for (closed-set) SSL may perform better in the open-set scenario than what has been previously suggested by other works.

Furthermore, this work proposes another method for OSSL: the Beta-model. This method proposes a novel score for ID/OOD classification and introduces the use of the expectation-maximization algorithm in OSSL, for estimating conditional distributions of scores given ID or OOD data. This method demonstrates state-of-the-art results on many benchmark problems for OSSL.

Keywords: Semi-supervised learning, open-set semi-supervised learning, deep learning, classification.

List of Publications

This thesis is based on the following publications:

[A] **Erik Wallin**, Lennart Svensson, Fredrik Kahl, Lars Hammarstrand, "DoubleMatch: Improving Semi-Supervised Learning with Self-Supervision". Published at ICPR 2022.

[B] **Erik Wallin**, Lennart Svensson, Fredrik Kahl, Lars Hammarstrand, "Improving Open-Set Semi-Supervised Learning with Self-Supervision". Manuscrupt 2023.

[C] **Erik Wallin**, Lennart Svensson, Fredrik Kahl, Lars Hammarstrand, "Beta-model: Open-Set Semi-Supervised Learning with In-Distribution Subspaces". Manuscrupt 2023.

Acknowledgments

I would like to extend my heartfelt gratitude to my main supervisor, Lars Hammarstrand, for his support and invaluable guidance throughout this firt part of my academic journey. I am also deeply thankful to my academic co-supervisors, Lennart Svensson and Fredrik Kahl, for their valuable contributions and assistance.

Moreover, I am truly grateful for the support and opportunities provided by Saab AB. I would like to express my special thanks to my insdustrial supervisors, Håkan Warston, Patrik Dammert, and Albert Nummelin for their mentorship and expertice. Additionally I am thankful to my manager, Per Gustavsson, for his encouragement and support.

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

The computations required for this thesis were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) and the Swedish National Infrastructure for Computing (SNIC), partially funded by the Swedish Research Council through grant agreements no. 2022-06725 and no. 2018-05973.

Acronyms

SSL:	Semi-supervised learning
OSSL:	Open-set semi-supervised learning
AUROC:	Area under receiver-operator curve
OSR:	Open-set recognition
EMA:	Exponential moving average
ID:	In distribution
OOD:	Out of distribution

Contents

Part I

Overview

CHAPTER 1

Introduction

In recent years, the field of deep learning has experienced a substantial surge, with applications spanning across a wide range of domains. Tasks in machine learning, such as image classification [1], image segmentation [2], generative modeling [3], language modeling [4], and audio processing [5], have been profoundly impacted by the advancements in deep learning techniques. This progress is continually evolving, driven by the expansion of datasets, improved compute power, and refined methods.

For a significant duration, the achievements in the field of deep learning relied on the framework of supervised learning. In supervised learning, a model is fed training data in which each sample is accompanied with a ground-truth label. The objective is to optimize the model and learn a mapping that aligns the input data with the corresponding labels. The goal of this optimization is that learned model generalizes to new, unseen data such that it can be deployed in real-world scenarios.

Many breakthroughs in the realm of supervised learning can be credited to the expansion of large labeled datasets. Prominent examples include, *e.g.*, ImageNet [6], comprising over 14 million labeled images, or the text dataset SQuAD [7] that consists of over 100,000 question-answer pairs. However,

building these labeled datasets requires extensive efforts from human annotators, making the process non-scalable and expensive. Consequently, there is need for alternative approaches that can effectively leverage large amounts of available unlabeled data. Unlike labeled data, obtaining unlabeled data is generally considerably cheaper and they can be acquired through the means of web scraping or unsupervised sensor streams.

In contrast to supervised learning, *semi-supervised learning* offers methods for incorporating unlabeled data for model training. This approach involves combining labeled data with (many more) unlabeled data during the training process. Typically, the small labeled training set is provided by human annotators, which defines the problem we want to address. For example, the labeled set often contains the classes that we want our model to classify. The much larger unlabeled dataset contains valuable information that aids the model in improving performance on the task outlined by the labeled data. A common scenario is that we know that the unlabeled data contain the same classes as the labeled data, which can help us learn the distributions of these classes given some assumptions.

Methods for semi-supervised learning are typically driven by pseudo-labeling and consistency regularization. Pseudo-labeling involves letting a model trained on labeled data generate pseudo-labels for unlabeled data, to then incorporate these pseudo-labeled unlabeled data in the training process in a supervised manner. Consistency regularization, on the other hand, means encouraging consistent predictions given perturbations of unlabeled data. With these techniques, semi-supervised learning has achieved state-of-the-art results in ImageNet classification through incorporation of large extra unlabeled training sets [8]. Semi-supervised learning has also impressively been used to reach classification accuracies above 95% on CIFAR-10 [9] using only 40 labeled data (4 labels per class) [10].

There are however still many active research problems and open questions related to semi-supervised learning. One fundamental question is how to optimally utilize unlabeled data. While pseudo-labeling and consistency regularization have emerged as key components in semi-supervised learning, the exact implementations of these techniques remain an ongoing area of investigation [10]–[14]. Furthermore, recent works have been exploring the integration of techniques from self-supervised learning as a complementary learning signal from unlabeled data [15]–[17].

Another active research direction is semi-supervised learning with uncurated unlabeled datasets. Most methods for semi-supervised learning operate under the assumption that the labeled set and the unlabeled set share the same underlying distribution, in particular that they contain the same classes. However, in practice, since the advantage of using unlabeled data lies in its freedom from human vetting, we can rarely guarantee that the unlabeled set does not contain unknown classes, corrupted data, or outliers in other forms. These out-of-distribution (OOD) data may lead to performance losses when they appear in traditional methods for semi-supervised learning. Recently, there have been many works aimed at tackling this setting of semi-supervised learning, often denoted *open-set semi-supervised learning* (OSSL) [18]–[20].

A related problem is that of *class-imbalanced semi-supervised learning*. Another assumption that traditional methods for semi-supervised learning make is that both the labeled and the unlabeled sets are balanced in terms of classes. We can construct the labeled set so that it is balanced. However, as we do not know the labels of the unlabeled dataset, ensuring that the unlabeled dataset is balanced is challenging. This may cause issues since it is known that many methods tend to show bias towards majority classes when data are imbalanced [21]. Methods for handling class imbalances in semi-supervised learning is an active field [22], [23].

Finally, semi-supervised learning is often studied in the domain of computer vision. The main reason is the wide range of publicly available datasets for image classification that allow for easy evaluation and benchmarking of methods. This is, however, at the risk of methods becoming biased to this particular domain, not performing equally well for other modalities, such as audio, text, radar, and lidar. In particular, many current methods for semi-supervised learning in computer vision are reliant on domain-specific data augmentation [11], [12], that naturally are not easily transferable to other domains.

Contributions: This thesis includes three appended papers, whose main contributions lie in method development for semi-supervised learning. In Paper A, we propose DoubleMatch, a method for semi-supervised learning that aims to improve methods based on pseudo-labeling to more effectively utilize all unlabeled data. This is done by proposing the inclusion of a *self-supervised* loss component that is applied to all unlabeled data. Experimental results demonstrate that this added loss accelerates training speed and improves classification accuracy on various benchmark datasets. In Paper B, we

present SeFOSS, a method for the setting of open-set semi-supervised learning. SeFOSS builds on the DoubleMatch method by including the proposed self-supervision on all unlabeled data. Additionally, it uses an energy-based method for to determine if unlabeled data are in-distribution (ID) or out-of-distribution (OOD). The energy-based method for OOD detection is complemented with an adaptive procedure for determining a threshold to identify data that confidently belong to the known classes. The experimental results show that SeFOSS achieves strong and robust results across a wide range of open-set problems, both for classifying the known classes and for classifying data as in- or out-of-distribution. In Paper C, we look further into the problem of open-set semi-supervised learning. Paper C proposes a novel score for ID/OOD classification. Furthermore, Paper C introduces the use of the expectation-maximization algorithm in OSSL for estimating conditional distributions of scores given ID or OOD data. These proposed components are put together into the Beta-model that reaches state-of-the-art results on many benchmark problems for OSSL. Beyond the appended papers, this thesis offers overviews of semi-supervised learning for classification and open-set semi-supervised learning.

1.1 Thesis outline

This introductory chapter provides background information and sets the scope of the thesis. In the following chapter, we provide an overview of the field of semi-supervised learning for classification, focusing on methods applied in the domain of deep learning. In the third chapter, we cover the setting of open-set semi-supervised learning and provide a review of existing literature. The fourth chapter summarizes the appended papers. Finally, the thesis concludes with summarizing remarks and an outlook for future work.

CHAPTER 2

Semi-supervised learning for classification

Semi-supervised learning is a machine learning paradigm that lies between supervised and unsupervised learning. In this setting, training data consists of both labeled data and unlabeled data. The idea is to somehow leverage information from the unlabeled data, together with the typically much smaller labeled set, while training a model. Take for example the illustration in figure 2.1. Given only information from labeled data, we can form a reasonable classification boundary. However, with the added information from unlabeled data, a more accurate classification boundary can be inferred to lie between the two half-moons. This improved boundary would be difficult to determine using only the labeled data.

This chapter provides an overview of semi-supervised learning. While there exist works on semi-supervised learning for regression, this chapter focuses on the realm of classification. To establish the foundations, we start by presenting a formal problem definition. Subsequently, we cover the necessary assumptions that underlie methods for semi-supervised learning. We proceed to examine the historical progression of methods in the field of semi-supervised learning. In the latter and largest part of this chapter, we turn our attention to the application of semi-supervised learning in the context of deep learn-

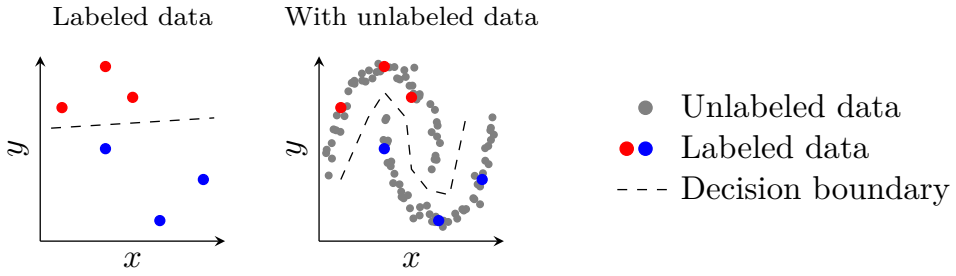

Figure 2.1: Illustration of semi-supervised classification with two classes (red and blue). With unlabeled data, we can better estimate the distributions of the two classes, and thus improve our decision boundary.

ing: this part of the chapter explores various techniques that are used for semi-supervised learning in the domain of deep learning.

2.1 Problem definition

In semi-supervised learning, we are provided with a labeled training set of independent and identically distributed data,

$$\{(x_i, y_i)\}_{i=1}^{m}; \qquad (x_i, y_i) \in \mathcal{X} \times \mathcal{Y}, \tag{2.1}$$

where $\mathcal{X} \subseteq \mathbb{R}^D$ is the input space with D being the input dimension, and $\mathcal{Y} = \{1, \ldots, C\}$ is the label space with C being the number of classes. These data have an underlying distribution $p(x, y)$. In addition to the labeled training set, we have set of independent and identically distributed unlabeled training data,

$$\{x_i\}_{i=m+1}^{n+m}; \qquad x_i \in \mathcal{X}, \tag{2.2}$$

where the underlying distribution $p(x)$ is the marginal distribution of $p(x, y)$.

The goal is to learn a mapping from the input space to the label space:

$$f_\theta : \mathcal{X} \to \mathcal{Y} \tag{2.3}$$

where f is parameterized by θ. This is typically achieved by minimizing the expectation of a risk function:

$$\underset{\theta}{\operatorname{argmin}} \underset{x,y \sim p(x,y)}{\mathbb{E}} \left[l(f_\theta(x), y) \right] + \alpha \underset{x \sim p(x)}{\mathbb{E}} \left[\Omega(x; \theta) \right], \tag{2.4}$$

where α is a scaling parameter to control the balance between the two terms. The expectation is often evaluated with Monte Carlo approximations using batches of the training data. The term for fitting the labeled training data is $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, which generally is implemented as a cross-entropy loss. The learning from unlabeled data occurs through the regularization term

$$\Omega : \mathcal{X} \to \mathbb{R}\,. \tag{2.5}$$

The construction of this regularization term is one of the key challenges in semi-supervised learning, as it defines how we utilize the unlabeled data for improving our learned model.

2.2 Assumptions

In order to learn from unlabeled data, we need to make some assumptions regarding the underlying structure of the data. The book *Semi-Supervised Learning* by Chapelle *et al.* [24] suggests three main assumptions in the form of the *smoothness assumption*, the *cluster assumption*, and the *manifold assumption*.

Smoothness assumption

The smoothness assumption can be briefly summarized by stating that if two points are close, then so should the corresponding outputs. Intuitively, this allows us to propagate information from our labeled training data to nearby unlabeled data. This assumption is necessary also for supervised learning, otherwise we cannot expect our model to generalize to unseen test data. Note that the *closeness* in this context is an open question, which we will return to in the manifold assumption.

Cluster assumption

In this assumption, we say that points that lie in the same cluster are likely to share class. This does not imply that each class consists of one cluster, but rather that each cluster is comprised of a single class. Equivalently, this assumption can be formulated as assuming that decision boundaries are expected to lie in *low-density regions*. An implication of this assumption is that

if we find a way to cluster our data, we can assign each cluster the class of any labeled training data contained in that cluster.

Manifold assumption

In the smoothness assumption, we say that points that are close should have outputs that are close. However, in high-dimensional spaces, pairwise distances between points tend to be non-expressive. Thus, we need to assume that the high-dimensional data lie on a low-dimensional manifold where we can compute more meaningful distances.

2.3 History of semi-supervised learning

The first instances of semi-supervised learning in the literature appeared in the 1960s and 1970s [25]–[27]. These methods employed a technique today called *self-training*, which involves an iterative process where the model is initially trained using only labeled data. In each subsequent step, model predictions on unlabeled data are used to expand the training set, and the model is retrained using the new training set. At this time, the methods were very general and were often referred to as *pattern recognition machines*.

In the 1990s, there was growing interest in more application-focused semi-supervised learning for text applications [28], [29]. Text is a typical domain where a lot of unlabeled data are available, but labeled data are expensive. For example, Yarowsky [28] used a form of self-training for semi-supervised sense classification of words.

2.4 Semi-supervised learning in deep learning

In the deep learning paradigm, input data are typically high-dimensional and our learned models are neural networks with many hidden layers. Naturally, many new techniques for semi-supervised learning have emerged to cater to this setting. This section covers some of the most popular techniques for semi-supervised in deep learning. Note that some details of the covered methods in this section may differ from the original works. The main purpose of this section is to give an overview of the general ideas and approaches of this paradigm.

Pseudo-labeling

One of the dominant techniques for semi-supervised learning in deep learning is pseudo-labeling. This essentially means using model predictions on unlabeled training data as training labels. A simple early version of this technique was introduced by the pseudo-label method [30]. The pseudo-label method simply takes the class with the largest predicted probability for each unlabeled sample and uses this as the training label. Sticking to the notation from (2.5), letting Ω be an element-wise loss for unlabeled data, we obtain

$$\Omega^{\text{pseudo label}} = H\left(\underset{y'}{\text{argmax}}[p_\theta(y'|x)], p_\theta(y|x)\right), \tag{2.6}$$

where $p_\theta(y|x)$ is the predicted distribution over classes for unlabeled sample x and argmax$[\cdot]$ is defined as $\mathbb{R}^C \to \mathbb{R}^C$ so that it returns a one-hot vector where the position for the one corresponds to the position of the largest element of the input vector. The cross entropy, $H(\cdot, \cdot)$, is calculated between two discrete probability distributions, p^a and p^b, as

$$H(p^a, p^b) = -\sum_{i=1}^{C} p_i^a \log p_i^b, \tag{2.7}$$

where p_i^a and p_i^b are the i-th elements of $p^a \in \mathbb{R}^C$ and $p^b \in R^C$, respectively.

It has later been found that using only pseudo-labels for data with confident model predictions tends to yield better results. For example, FixMatch [12] and UDA [11] assign pseudo-labels to unlabeled data that satisfy

$$\max_y p_\theta(y|x) > \tau, \tag{2.8}$$

where τ is the confidence threshold. This results in a loss on unlabeled data that looks similar to

$$\Omega^{\text{FixMatch}} = \mathbb{1}\{\max_{y'} p_\theta(y'|x) > \tau\} H\left(\underset{y'}{\text{argmax}}[p_\theta(y'|x)], p_\theta(y|x)\right), \tag{2.9}$$

where $\mathbb{1}(\cdot)$ is the indicator function.

A more involved pseudo-labeling procedure is proposed in the method Meta Pseudo Labels [8]. This method uses an alternating procedure where a stu-

dent model is updated based on the pseudo-labels it is provided by a teacher model. The teacher model on the other hand is updated based on the student performance on a held-out validation set. Intuitively, this can be interpreted as optimizing the teacher model to produce the best possible teaching samples, hence the name Meta Pseudo Labels.

Relating the pseudo-labeling technique back to our assumptions of section 2.2, we can interpret this as an application of the clustering assumption. When we train our model to produce confident predictions in high-density regions (regions where our training data are located), we are implicitly pushing the decision boundaries to low-density regions in accordance with the clustering assumption.

Adaptive and dynamic thresholds

The pseudo-labeling procedure of FixMatch and UDA, as described in (2.8), relies on a static threshold. Many recent works have focused on replacing this static threshold with dynamic and adaptive thresholds. This direction of research is motivated by two main factors. Firstly, the varying learning difficulty associated with different classes incentivizes using class-dependent thresholds. For example, the model may produce less confident predictions for a particular class, causing fewer pseudo-labels and hindering learning for that class. Secondly, neural networks tend to generate increasingly confident predictions as the training progresses, suggesting that thresholds can be modified based on the completed number of training steps.

One example of a method that proposes a dynamic confidence schedule as function of the training time is Dash [13]. This work identifies that FixMatch tends to produce very few pseudo-labels early in training, but also increasingly many *incorrect* pseudo-labels in the later stages of training. To counteract this, Dash, suggests a schedule for the threshold that decreases monotonically as training progresses. The dynamic threshold is computed as

$$\tau_t^{\text{Dash}} = C\gamma^{-(t-1)}\hat{p}, \tag{2.10}$$

where t is the current timestep in training, C and γ are a constant hyperparameters, and $\hat{p}$ is the base threshold that is computed based on a pre-training phase.

A method that instead proposes adaptive thresholds per class, *e.g.*, is Flex-

Match [14]. FlexMatch adjusts the class-dependent thresholds depending on how many pseudo-labels are assigned to each class: if a class is less frequent in the pseudo-labels, its threshold is lowered in order to assign more pseudo-labels corresponding to that particular class. A similar method that also uses class-dependent adaptive thresholds is FreeMatch [10]. In FreeMatch, the class-dependent thresholds are computed based on the average prediction confidence for each class: classes that are less confidently predicted are assigned lower thresholds.

Consistency regularization

Another major technique for semi-supervised learning in deep learning is consistency regularization. The idea of consistency regularization is to minimize the difference in predictions for similar data points. Similar data points are often generated by applying perturbations to the training data. Given original unlabeled training sample, x, and the corresponding perturbation, $\tilde{x}$, the general structure for consistency regularization looks like

$$\Omega^{\text{Consistency regularization}} = d\left(f_\theta(x), f_\theta(\tilde{x})\right), \tag{2.11}$$

where $d(\cdot, \cdot)$ is some distance measure, *e.g.*, mean squared error or KL divergence. The distance may also be calculated between two different perturbations, instead of the original data and a single perturbation.

Consistency regularization relates both to the smoothness assumption and the cluster assumption of section 2.2. The smoothness assumption is addressed by manually constructing close inputs through perturbations, to then encourage similar predictions for these nearby inputs. Additionally, the consistency regularization is applied mainly in high-density regions due to the concentration of training data in these regions. This implicitly enforces similar predictions within clusters, which moves decision boundaries toward low-density regions, in accordance with the cluster assumption.

How perturbations for consistency regularization are designed has been an active field of research. An early version of consistency regularization in semi-supervised learning was used in the Ladder network [31], where perturbed data are created by injecting Gaussian noise to the activations at each layer of the network. The noisy activations are then denoised by a trainable decoder network. The resulting loss is the sum of squared errors between the denoised

activations and the activations from a clean pass through the network for all layers:

$$\Omega^{\text{Ladder networks}} = \sum_{l=1}^{L} \lambda_l \|z^l - \hat{z}^l\|^2 \tag{2.12}$$

where L is the number of layers in the network, λ_l is a layer-dependent scaling factor, z^l are the clean activations for unlabeled sample x at layer l, $\hat{z}^l$ are the corresponding denoised activations, $\|\cdot\|$ is the l^2 norm.

The subsequent Π-model [32] instead applies consistency regularization directly to the predicted probability distributions given two perturbations, $\hat{p}^a_\theta(y|x)$ and $\hat{p}^b_\theta(y|x)$. The perturbations are obtained by applying two instances of some stochastic data augmentation on x along with two different realizations of the stochastic *dropout* regularization [33] in the forward pass through the neural network.[1] The obtained loss for unlabeled data is

$$\Omega^{\Pi\text{-model}} = \|\hat{p}^a_\theta(y|x) - \hat{p}^b_\theta(y|x)\|^2. \tag{2.13}$$

The concept of using moving averages as *teacher predictions* was introduced in the Temporal ensembling method [32]. With this terminology, the teacher prediction is typically treated as the ground truth for the student prediction. The basic motivation behind using a moving average as teacher prediction is to generate less noisy targets. Temporal ensembling uses the same perturbation strategy as the Π-model. However, instead of using two different perturbations, the teacher prediction is an exponential moving average of the perturbed student prediction, updated each epoch, [2] as

$$p^{\text{teacher}}_\theta(y|x) \leftarrow \beta p^{\text{teacher}}_\theta(y|x) + (1-\beta) p^{\text{student}}_\theta(y|x), \tag{2.14}$$

where β is the momentum parameter (typically close to, but smaller than 1). The loss is then given by

$$\Omega^{\text{Temporal ensembling}} = \|p^{\text{teacher}}_\theta(y|x) - p^{\text{student}}_\theta(y|x)\|^2 \tag{2.15}$$

The method Mean teacher [34] develops the idea of using moving averages

[1] Dropout is a common regularization technique for neural networks that involves stochastically masking neurons and their connections in each forward pass during training.

[2] An epoch in this context means the time it takes to cycle through the full training set in the training process.

as teacher predictions by taking an exponential moving average of the model parameters. This has the advantage that the exponential moving average can be updated every training step instead of once every epoch. The average of the model parameters are updated each training step as

$$\theta_{\text{EMA}} \leftarrow \beta\theta_{\text{EMA}} + (1-\beta)\theta. \tag{2.16}$$

Mean teacher uses a perturbation strategy that consists of a data augmentation, Gaussian noise on the input layer, and dropout. The perturbation is applied both to the teacher prediction and the student prediction (in two different realizations). The resulting loss is

$$\Omega^{\text{Mean teacher}} = \|\hat{p}_{\theta_{\text{EMA}}}(y|x) - \hat{p}_{\theta}(y|x)\|^2 \tag{2.17}$$

The methods covered so far in this section are relying on random perturbations for consistency regularization, *i.e.*, these methods smooth the prediction function in random directions around the input. The method Virtual adversarial training (VAT) [35] takes another approach. In VAT, the idea is to smooth the prediction function in the *least* smooth direction with respect to the input, *i.e.*, the *adversarial direction*. The adversarial direction is, in this context, defined as the direction of the point, within a small region of the input, that gives the largest change in prediction (relative to the unperturbed input). Formally, the loss is written as

$$\Omega^{\text{VAT}} = d_{\text{KL}}\left(p_\theta(y|x), p_\theta(y|x + r_{\text{adv}})\right), \tag{2.18}$$

where

$$r_{\text{adv}} = \underset{r;\|r\|<\epsilon}{\operatorname{argmax}}\, d_{\text{KL}}\left(p_\theta(y|x), p_\theta(y|x+r)\right). \tag{2.19}$$

Here, $d_{\text{KL}}(\cdot,\cdot)$ is the KL-divergence and ϵ is a small scalar that sets the size for the region in which we look for the adversarial direction. Unfortunately, there exists no closed form expression for r_{adv}, so VAT uses a one-step power iteration to approximate r_{adv}.

Data augmentation

Early implementations of consistency regularization often relied on simple techniques for data augmentation, such as horizontal flips and translations in

the context of images. However, notable achievements were made with the introduction of optimized domain-specific augmentations in ReMixMatch [36], FixMatch [12], and UDA [11]. These augmentations are, *e.g.*, RandAugment [37] for images, which comprises a set of operations, such as shearing, rotating, and adjusting colors or brightness. For a domain like language, these domain-specific augmentations can be, *e.g.*, back-translation [38] that involves translating a sentence from language A to language B, and then back to language A, to obtain a slightly perturbed version of the original sentence. Notably, ReMixMatch and FixMatch pioneered a setup of using weak augmentations for teacher predictions and strong augmentations for student predictions in the image domain. The weak augmentation consists of a horizontal flip and translation and the strong augmentation consists of Cutout [39], followed by two randomly sampled operations from RandAugment. Examples of these weak and strong augmentations can be seen in figure 2.2. The augmentation strategy of ReMixMatch and FixMatch has been widely adopted by many subsequent works [10], [13], [14], [16], [17], [40], [41].

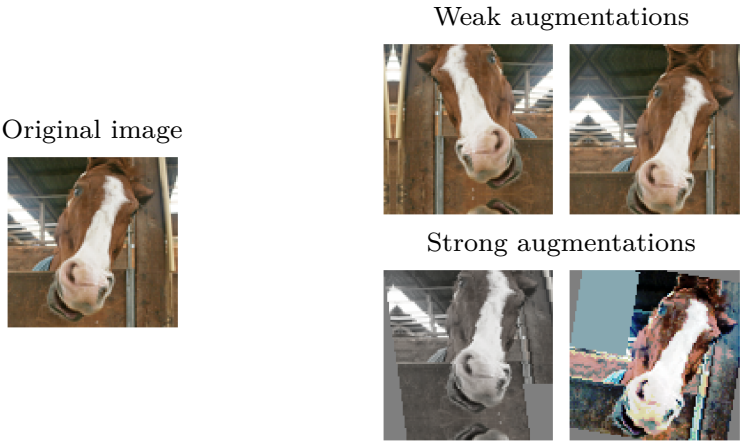

Figure 2.2: The currently widely used augmentation strategies for semi-supervised learning, consisting of weak and strong augmentations of images. Weak augmentations are horizontal flips and stochastic translations. Strong augmentations comprise operations such as Cutout, shearing, rotations, and color filters.

Interpolation consistency

Another form of data augmentation is to use interpolations of training data. This strategy was introduced for supervised learning under the name *mixup* [42]. The idea is to create new training data by interpolating both input data and corresponding labels using the Mix-operation, defined as

$$\mathrm{Mix}_\lambda(a, b) = \lambda a + (1 - \lambda)b, \tag{2.20}$$

where λ is a parameter between 0 and 1 that is sampled from a Beta distribution. The methods Interpolation consistency training (ICT) [43] and MixMatch [44] introduced the idea of using interpolations in semi-supervised learning. For unlabeled data, we cannot interpolate labels to form optimization targets, instead we can interpolate model predictions. For example, ICT uses the exponential moving average of the model parameters to form targets for interpolations of unlabeled data according to

$$\Omega^{\mathrm{ICT}}(x_a, x_b) = \|p_\theta(y|\mathrm{Mix}_\lambda(x_a, x_b)) - \mathrm{Mix}_\lambda(p_{\theta_{\mathrm{EMA}}}(y|x_a), p_{\theta_{\mathrm{EMA}}}(y|x_b))\|^2, \tag{2.21}$$

where x_a and x_b are two different unlabeled samples. Training with interpolations can be argued being well-aligned with the cluster assumption of section 2.2. If we are considering a classification problem with more than a few classes, it is likely that that x_a and x_b belong to different classes, and thus different clusters. Assuming x_a and x_b are not incorrectly predicted as the same class, the interpolation loss will move the decision boundary toward the region between these data, which is a low-density region.

Self supervision

A related field to semi-supervised learning is *self-supervised learning*. In self-supervised learning, we are training a model using training data fully without labels. The goal is not to learn a classifier, but to learn a useful low-dimensional representation of the often high-dimensional data. Note how this relates to the manifold assumption of section 2.2. There are various techniques that are commonly used for self-supervised learning. One is to enforce prediction consistency across augmentations of data (much like consistency regularization for semi-supervised learning) [45]–[48]. Another technique involves training the model to reconstruct masked regions of input data [49],

[50]. Additionally, a common approach is to train the model to perform a pretext task, such as predicting the angle of a stochastic rotation applied to training images [51]–[53].

Influential works for self-supervised learning in the image domain made use of so called contrastive learning [45], [46], which means not only enforcing similar predictions for different versions of the same data, but also increasing the disagreement of representations given different data. One argument for the contrastive loss is that without enforcing the disagreements, the model can converge to the collapsed solution: predicting the same representation for all data. However, subsequent works [47], [48], [54] found that collapse can be avoided without contrastive learning by instead using exponential moving average as teacher models and by the use of cleverly placed stop gradient operations.

There are many works that borrow techniques from self-supervised learning for semi-supervised learning. The motivation is that the self-supervision can improve the latent representations of data in the model, or that it can help methods based on confidence-based pseudo-labeling (see (2.9)) to utilize all unlabeled data, not only data that fall above the confidence threshold.

One work that incorporates techniques from self-supervised learning for semi-supervised learning is S4L [15]. S4L employs a rotation loss to unlabeled data, formulated as follows:

$$\Omega^{\mathrm{S4L}} = \frac{1}{4} \sum_{r \in \mathcal{R}} H(r_{\mathrm{target}}, g(z_r)), \tag{2.22}$$

where $\mathcal{R} = \{0°, 90°, 180°, 270°\}$ and r_{target} is the one-hot vector that denotes the current rotation, *e.g.*, $r_{\mathrm{target}} = (0, 1, 0, 0)^T$ for $r = 90°$. Here, z_r is the latent representation of the network (predicted by some backbone model f_θ) for an unlabeled image x that has undergone rotation r, while g is a trainable 4-way classifier that predicts the rotation based on the latent representation. The rotation prediction serves as a typical pretext task, since the main interest lies in improving the latent representations. By creating latent representations that can be used for predicting rotations, they are hopefully also more useful for classifying the actual classes of our training set.

EnAET [55] similarly employs a self-supervised pretext task for semi-supervised learning. However, instead of predicting rotations, the model is trained to predict the continuous parameters of more general transformations such as

projective and affine transformations.

In, DoubleMatch [56], the idea is to improve the utilization of unlabeled data in methods that employ confidence-based pseudo-labeling. To enable learning from all unlabeled data, DoubleMatch proposes an auxiliary self-supervised loss to all unlabeled data to align the latent representations for weak and strong augmentations of a given unlabeled image, given by

$$\Omega^{\text{DoubleMatch}} = -\frac{z_w \cdot g(z_s)}{\|z_w\| \cdot \|g(z_s)\|}, \tag{2.23}$$

where z_w and z_s are the latent representations for weak and strong augmentations of unlabeled image x, respectively. The trainable linear transformation $g(\cdot)$ is used to map the latent representations of strongly augmented data to the latent space of weak augmentations.

Recently, CCSSL [40], SimMatch [17], and ProtoCon [16] have used forms of contrastive learning for latent representations where pseudo-labels are used for determining which data to pull together and pull apart.

CHAPTER 3

Open-set semi-supervised learning

In semi-supervised learning, it is commonly assumed that labeled and unlabeled training data follow the same distribution and that the set of classes for the labeled and the unlabeled training sets are equal. For many practical applications, this assumption is probably not reasonable. On the contrary, it seems natural to assume that the unlabeled set may contain outliers, unseen classes, or corrupted data. In this case, we want to make sure that these out-of-distribution (OOD) data do not harm the training of our model, and perhaps also that our model can learn to identify the OOD data at test time. Take figure 3.1 as an example. Here, the unlabeled data give us information about the distributions of class A and class B, but they also indicate the existence of a third class that is not present in our labeled training data. A well-trained model on these data preferably has the ability to classify class A and class B, but also to identify data that likely do not belong to class A or class B.

This chapter gives an overview of the field of open-set semi-supervised learning. Note that some works denote this field safe semi-supervised learning, or robust semi-supervised learning. We start by expanding the problem formulation from chapter 2 to fit the open-set problem. Next, we cover existing

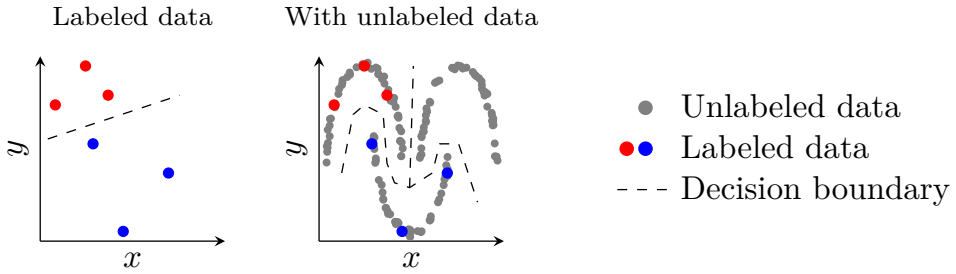

Figure 3.1: Illustration of open-set semi-supervised learning. The unlabeled data can improve our estimations of the class distributions, but they also indicate the presence of an unknown class. A preferable decision boundary is to classify red and blue based on the two leftmost half-moons, but also to reject the unknown class belonging to the rightmost half-moon.

methods and techniques that attempt to tackle this problem. Finally, we mention a few related research problems.

3.1 Problem formulation

Similarly to the problem formulation presented for the closed-set setting in chapter 2, we have a labeled training set

$$\{(x_i, y_i)\}_{i=1}^{m}; \qquad (x_i, y_i) \in \mathcal{X}_l \times \mathcal{Y}_l, \tag{3.1}$$

where again $\mathcal{X}_l \subseteq \mathbb{R}^D$ with D being the input dimension, and $\mathcal{Y}_l = \{1, \ldots, C\}$. We assume our labeled samples are independent and identically distributed from an underlying distribution $p_l(x, y)$. Additionally, we have the unlabeled training set

$$\{x_i\}_{i=m+1}^{n+m}; \qquad x_i \in \mathcal{X}_{ul}, \tag{3.2}$$

such that $\mathcal{X}_l \subseteq \mathcal{X}_{ul} \subseteq \mathbb{R}^D$, and the corresponding (unknown) labels associated with the unlabeled samples are in $\mathcal{Y}_{ul} = \{1, \ldots, C, C+1, \ldots, C+K\}$, meaning there are K novel classes in the unlabeled set that are not part of the labeled set. We assume that our unlabeled samples are independent and identically distributed with the underlying distribution $p_{ul}(x)$. Note that, in contrast to chapter 2, we no longer assume that $p_{ul}(x)$ is the marginal distribution of $p_l(x, y)$.

In general, we are interested in learning the classification mapping corresponding to our labeled training set:

$$f_\theta : \mathcal{X}_l \to \mathcal{Y}_l \tag{3.3}$$

However, we may also be interested in the binary classification of in-distribution (ID) and out-of-distribution (OOD) data:

$$P(y \leq C|x); x \in \mathcal{X}_{ul}, \tag{3.4}$$

meaning predicting the probability of a sample belonging to the known classes, given a sample from the unlabeled distribution.

Taking the classification of data as ID or OOD one step further, we can also consider unknown classes that are fully unseen during training, *i.e.*, not part of the unlabeled set. These classes can be introduced by the test set X_{test}, such that $X_l \subseteq X_{\text{test}} \subseteq \mathbb{R}^D$ where the corresponding classes belong to $\mathcal{Y}_{\text{test}} = \{1, \ldots, C, C+K+1, \ldots, C+K+L\}$, meaning the unknown classes from the unlabeled training set are replaced by new unknown test classes. We are in this case interested in modeling

$$P(y \leq C|x); x \in \mathcal{X}_{\text{test}}. \tag{3.5}$$

Different works in open-set semi-supervised learning focus on different goals. Some works primarily aim to achieve high closed-set accuracy, meaning attaining high accuracy on the known classes. This corresponds to having a well performing closed-set classifier, f_θ, as defined in (3.3). These works argue that unknown classes in the unlabeled training set can harm the closed-set performance of traditional methods for semi-supervised learning.

Other works place greater emphasis on open-set recognition, which involves the ability to distinguish known classes from unknown classes. The motivation for these works is that if unknown classes appear at training, it is unlikely that test data comprise only the known classes. Open-set recognition can be either in the form of distinguishing the known classes from the unknown classes in the unlabeled training set, as represented in (3.4), or it can be in the form of identifying known classes in the presence of classes completely unseen during training, as described in (3.5).

The training objective of open-set semi-supervised can generally be written

as

$$\underset{\theta}{\operatorname{argmin}} \underset{x,y\sim p_l(x,y)}{\mathbb{E}} [l(f_\theta(x), y)] + \alpha \underset{x\sim p_{ul}(x)}{\mathbb{E}} [\Omega(x;\theta)], \tag{3.6}$$

which is similar to the objective of the closed-set case (see (2.4)), with the difference that the unlabeled term now is an expectation over the distribution that may contain unknown classes, $p_{ul}(x)$.

3.2 Existing techniques for open-set semi-supervised learning

This chapter covers existing techniques for open-set semi-supervised learning. We try to categorize methods based on what kind of technique it employs, in an attempt to summarize existing approaches and research directions in this field.

Filtering in-distribution data from unlabeled data

A recurring theme in methods for open-set semi-supervised learning is to clean the unlabeled data by attempting to identify which data belong to the known classes and which do not. When in-distribution (ID) data are identified, these can be used in the unsupervised loss of a traditional SSL method. How to best identify which data belong to the known or unknown classes is however still an open question.

Many methods [57]–[59] resort to different forms of the softmax confidence score,

$$\max_y p_\theta(y|x), \tag{3.7}$$

where the idea is that ID data yields higher-confidence predictions than OOD data. For example, UASD [57] uses the confidence of the average prediction from the most recent epochs given an unlabeled sample:

$$c(x) = \max_y \frac{1}{t} \sum_{i=1}^{t} p_{\theta_i}(y|x), \tag{3.8}$$

where $p_{\theta_i}(y|x)$ for $i = 1, \ldots, t$ are the network predictions for sample x from the t most recent epochs during training. A sample is classified as ID if

$c(x) > \tau$, where τ is set as the average confidence given a labeled ID validation set.

MTCF [60] takes a different approach by employing a separate prediction head, $s(x)$, responsible for the binary prediction of ID or OOD. The parameters, θ, of the extra prediction head (and the rest of the model) are jointly optimized with the (unknown) scalar scores $s_i \in [0, 1], i = m + 1, \ldots, m + n$ for all unlabeled data. In practice, this results in an alternating procedure as follows:

1. Update θ by minimizing some semi-supervised loss using only unlabeled data predicted as ID, and a binary cross-entropy loss for classifying data as ID or OOD,

2. Reassign ID scores for unlabeled data $s_i \leftarrow s(x_i)$ for $i = m+1, \ldots, n+m$.

The ID scores for labeled training data are fixed as $s_i = 1, i = 1, \ldots, n$.

In OpenMatch [18], ID data are identified by resorting to one-vs-all classifiers. With one-vs-all classifiers, there is one prediction head for each known class, responsible for predicting if a sample belongs to that particular class or any other class. A sample is identified as OOD if none of the one-vs-all classifiers gives a high-confidence prediction. An advantage of using one-vs-all classifiers for OOD detection is that each classifier has access to both positive and negative labeled training data from the labeled training set.

SeFOSS [61] uses the free-energy score, as proposed by [62], to classify data as ID and OOD. The free-energy score is obtained by

$$E(x) = -T \cdot \log \sum_{i=1}^{C} e^{f_{\theta,i}(x)/T} \tag{3.9}$$

where T is a scalar hyperparameter and $f_{\theta,i}(x)$ is the predicted logit[1] associated with class i. Knowing that the network is trained to produce large logits for ID data (through the cross-entropy loss), we can assume that the free energy generally takes larger negative values for ID data than for OOD data. While still being easy and cheap to compute, the free-energy score tends to produce better results for OOD detection than the softmax confidence [62].

[1] The logits are the final network activations before they are transformed to a probability distribution by the softmax function.

SAFE-STUDENT [63] builds on the free-energy score and proposes energy discrepancy for OOD detection. The motivation is that because the free-energy score is dominated by the largest logit through the exponential function, it does not sufficiently take into account information contained in other logits. The proposed energy discrepancy is approximately given by

$$ED(x) \approx f_{\theta,y_{\max}}(x) - f_{\theta,y'}(x) \tag{3.10}$$

where

$$y_{\max} = \underset{y \in \mathcal{Y}_l}{\operatorname{argmax}} f_{\theta,y}(x) \text{ and } y' = \underset{y \in \mathcal{Y}_l \setminus y_{\max}}{\operatorname{argmax}} f_{\theta,y}(x). \tag{3.11}$$

This means that $ED(x)$ is approximately equal to the difference between the largest and the second largest logits, which should be large for ID data.

Recycling of out-of-distribution data

Another idea that appears in the literature for open-set semi-supervised learning is to "recycle" OOD data. The goal is to identify useful information in OOD data that somehow can be used in the training process to improve model performance.

One of these methods is TOOR [59]. TOOR attempts to recycle OOD data by closing the distribution gap between the features of ID data and the features of OOD data. This is done by adversarial training of an feature extractor, $F(\cdot)$, parameterized by θ_F, together with a discriminator, $D(\cdot)$, parameterized by θ_D. The feature extractor is responsible for extracting features from data samples and the discriminator classifies data as ID or OOD based on their features. The adversarial objective is

$$\min_{\theta_F} \max_{\theta_D} \mathop{\mathbb{E}}_{x \sim p_{\text{OOD}}(x)} \log D(F(x)) + \mathop{\mathbb{E}}_{x \sim p_{\text{ID}}(x)} \log(1 - D(F(x))). \tag{3.12}$$

With this objective, the discriminator is trained to correctly classify data as ID or OOD, but the feature extractor is trained to fool the discriminator by making OOD features indistinguishable from ID features. By transforming OOD features to the ID space, the idea is that the OOD data can be used to improve learning for the classification problem on $\mathcal{Y}_l$ by inclusion in, *e.g.*, a consistency regularization or pseudo-labeling loss.

OSP [20] uses OOD data to "prune" OOD features from ID data. This is

done by matching ID data with OOD samples that has similar features. The features of the ID sample are then transformed by subtracting a vector that is parallel to the OOD features. The model is then forced to focus on the ID features by encouraging similar predictions for the ID data before and after semantic pruning.

Another technique related to recycling is Style disturbance [64]. Style disturbance takes inspiration from the idea of neural style transfer which takes the content of one sample and transforms it to the style of another sample. Style disturbance expands the training set of open-set semi-supervised learning by creating new data with the contents of ID data and the styles of OOD data by employing style transfer with AdaIN [65].

Robust optimization

Another line of research for open-set semi-supervised learning attempts to adjust the optimization steps so that parameters updates never harm performance on ID data. Some of these methods resort to bi-level optimization. For example, DS3L [66] and WRSSL [67] use bi-level optimization to weight unlabeled data such that the resulting updates minimize a supervised loss on a labeled training set. For example, DS3L learns a weighting function $w_\alpha(\cdot)$, parameterized by α, that is used to weight each unlabeled sample in a traditional SSL loss. The bi-level optimization objective can be written as

$$\min_{\alpha} \mathop{\mathbb{E}}_{x,y\sim p_l(x,y)} \left[l(y, p_{\hat{\theta}}(y'|x)) \right] \tag{3.13}$$

such that

$$\hat{\theta} = \operatorname*{argmin}_{\theta} \mathop{\mathbb{E}}_{x,y\sim p_l(x,y)} \left[l(y, p_{\hat{\theta}}(y'|x)) \right] + \mathop{\mathbb{E}}_{x\sim p_{ul}(x)} \left[w_\alpha(x)\Omega(x;\theta) \right], \tag{3.14}$$

where $l(\cdot,\cdot)$ and $\Omega(\cdot)$ are the loss functions for labeled and unlabeled data, respectively. The intuition behind this objective is that the model parameters are learned by the inner weighted SSL objective, but the weighting function, learned through the outer objective, makes sure that the weighting of the unlabeled data causes the inner objective to be aligned with performance on ID data.

SPL [68] similarly uses bi-level optimization for robust optimization. However, instead of learning a weighting function for unlabeled data, it optimizes

a mask for the model parameters. The idea is to find the parameters that are associated with features corresponding to ID data to restrict negative effects from OOD data.

A different approach for robust optimization is proposed in Fix-a-Step [69]. The idea of Fix-a-Step is to ignore the gradient from unlabeled data if the gradient from unlabeled data does not point in a similar direction as the gradient for labeled data. Given the gradient associated with the labeled loss, $g^L = \nabla_\theta l$, and the gradients associated with the unlabeled loss, $g^U = \nabla_\theta \Omega$, the parameter updates in each training step is

$$\theta \leftarrow \begin{cases} \theta - \epsilon(g^L + \alpha g^U), & \text{if } g^L \cdot g^U > 0 \\ \theta - \epsilon g^L & \text{otherwise,} \end{cases} \tag{3.15}$$

where ϵ is the learning rate and α is a scaling for the unlabeled loss. The intuition behind this procedure is that if the inner product between the gradients is positive, $g^L \cdot g^U > 0$, the angle between the gradients is less than $90°$ and we thus assume that the unlabeled loss is somewhat aligned with the labeled loss. However, if the inner product is negative, the gradients are pointing in different directions, so the gradients from unlabeled data can potentially harm performance on labeled data. In this case, we ignore the gradient from unlabeled data.

Self-supervision

Some works for open-set semi-supervised learning relax the idea of learning primarily from unlabeled data that are ID. These works instead incorporate techniques from self-supervision to learn from all unlabeled data, regardless of whether they are ID or OOD. This is to avoid limiting learning to samples confidently ID, recognizing that OOD data may possess valuable information that can contribute to performance on ID data as well.

Some examples are T2T [70] and OSP [20] that employ the rotation loss of (2.22) on all unlabeled data. Another example is SeFOSS [61] that takes the self-supervision proposed by DoubleMatch [56] and applies it for open-set semi-supervised learning.

OpenCOS [71] suggest self-supervised contrastive pre-training using all training data. The pre-trained model is then used to detect ID and OOD data from the unlabeled training set. The labeled data and the ID data from the unla-

beled set can then be used to fine tune the pre-trained model using some SSL method.

In Υ-model [58], the authors find that if we somehow know the true classes and labels of the OOD data, these classes can be added to the classification problem with the effect of increased accuracy on the ID classes. Phrased differently, if we learn to classify the entire $\mathcal{Y}_{ul}$, we can improve the performance on $\mathcal{Y}_l$. Motivated by this finding, the Υ-model performs deep clustering [72] on OOD data to identify these unknown classes and include them in the classification problem. The number of unknown classes, K, is however a hyperparameter and it may not be possible to know the number of unknown classes in the OOD data.

3.3 Related research problems

There are some research problems that are closely related to that of open-set semi-supervised learning. One is open-world semi-supervised learning [73]–[77]. In open-world semi-supervised learning, the focus lies in discovering and classifying the unknown classes of the unlabeled data. In contrast to open-set semi-supervised learning that generally focuses on classification accuracy on $\mathcal{Y}_l$, open-world semi-supervised learning aims to achieve high accuracy on $\mathcal{Y}_{ul}$.

Another related topic is open-set domain adaptation [78], [79]. In domain adaptation, we generally consider a labeled source domain and want to adapt to a unlabeled target domain containing the same classes as the source domain, but with a domain shift. In open-set domain adaptation, it is assumed that the target domain contains unknown classes. We can see that this problem becomes similar to open-set semi-supervised learning if we consider the labeled source domain as our labeled training set, and the unlabeled (open) target domain as our unlabeled training set.

In semantically coherent out-of-distribution detection [80], [81], the goal is to classify unseen data as ID or OOD. Many other works on OOD detection simply consider one dataset as ID and another dataset as OOD, even if they contain classes that are semantically very close. In semantically coherent OOD detection, the aim is to identify classes in the unseen dataset that are semantically close to the known dataset and identify these as ID (*e.g.*, dogs in the unseen dataset should be considered ID if there are dogs in the known dataset). Models for semantically coherent OOD detection are gen-

erally trained using a labeled dataset as ID data, and an unlabeled dataset that contains classes that are semantically close to the ID data and some classes that are not. This setup is similar to that of open-set semi-supervised learning.

CHAPTER 4

Summary of included papers

This chapter provides a summary of the included papers.

4.1 Paper A

Erik Wallin, Lennart Svensson, Fredrik Kahl, Lars Hammarstrand
DoubleMatch: Improving Semi-Supervised Learning with Self-Supervision
Published in proceedings of the 2022 26th International Conference on Pattern Recognition (ICPR),
pp. 2871-2877
 DOI: 10.1109/ICPR56361.2022.9956182 .

This paper proposes a method for (closed-set) semi-supervised learning. A common technique for existing methods for semi-supervised learning is to employ confidence-based pseudo-labeling on unlabeled data. This process assigns artificial labels to unlabeled data for which the model's predictions exceed a confidence threshold. Data for which the model produces less confident predictions are disregarded from the training objective. Consequently, these methods may ignore large parts of unlabeled data, in particular for more

challenging classification problems. For better utilization of unlabeled data, this paper proposes the inclusion of a self-supervised component to enable learning from all unlabeled data. This additional self-supervision is applied to all unlabeled data and involves aligning feature predictions across weak and strong augmentations of each sample. More specifically, we implement this self-supervision as an extension of the widely adopted SSL baseline FixMatch. Our proposed method is evaluated benchmark datasets CIFAR-10, CIFAR-100, SVHN, and STL-10. DoubleMatch demonstrates particularly strong results on CIFAR-100 and STL-10, with improved accuracies and training speed when comparing to FixMatch. However, on the relatively simpler classification tasks of CIFAR-10 and SVHN, our proposed method is not equally effective. A possible explanation could be the model's ability to generate sufficiently many correct pseudo-labels when the classification problem is relatively straightforward, diminishing the benefits introduced by the additional self-supervision. **Contributions:** Erik Wallin did the main work. Lennart Svensson, Fredrik Kahl, and Lars Hammarstrand supervised.

4.2 Paper B

> **Erik Wallin**, Lennart Svensson, Fredrik Kahl, Lars Hammarstrand
> Improving Open-Set Semi-Supervised Learning with Self-Supervision
> *Manuscript 2023* .

This paper studies open-set semi-supervised learning (OSSL), a more realistic scenario where we assume that the unlabeled data may contain unknown classes not present in the labeled data. Many existing works for OSSL use methods that involve filtering out ID data from unlabeled data for inclusion in a traditional SSL loss. The method proposed in this paper, SeFOSS, instead follows the philosophy of DoubleMatch from Paper A, aiming to learn from all unlabeled data, regardless of them being ID or OOD. To achieve this, SeFOSS incorporates the self-supervision proposed by DoubleMatch on all unlabeled data. Additionally, SeFOSS applies a pseudo-labeling loss on unlabeled data that confidently belong to the known classes. To identify these confidently ID data, SeFOSS employs an energy-based method for discriminating between ID and OOD. To determine a threshold for assigning data as confidently ID, we propose an adaptive procedure based on the energy distribution of labeled data. SeFOSS is evaluated and compared with existing methods for OSSL

on open-set scenarios involving datasets CIFAR-10, CIFAR-100, SVHN, ImageNet, and noise. The experimental results show that SeFOSS exhibits an unmatched overall performance in terms of both closed-accuracy and OOD detection across the range of studied scenarios. While other methods perform well on a few scenarios, they fail to consistently and robustly perform on all scenarios. Moreover, this paper shows that methods for closed-set semi-supervised learning may perform better in terms of closed-set accuracy than previously reported by existing works. In fact, FixMatch outperforms all OSSL methods on closed-set accuracy in the experiments conducted in this paper. However, FixMatch performs poorly in terms of OOD detection, which is of significant importance for real-world applications.
Contributions: Erik Wallin did the main work. Lennart Svensson, Fredrik Kahl, and Lars Hammarstrand supervised.

4.3 Paper C

Erik Wallin, Lennart Svensson, Fredrik Kahl, Lars Hammarstrand
Beta-model: Open-Set Semi-Supervised Learning with In-Distribution Subspaces
Manuscript 2023 .

This paper proposes a method for open-set semi-supervised learning: Beta-model. In this method, we propose the use of a novel score for ID/OOD classification. This score is based on computing a subspace in feature space that is associated with ID data. For a test vector, the score is then computed as the cosine of the angle between this test vector and the subspace. Additionally, we propose to estimate the conditional distributions of scores for ID and OOD data. This is done through the use of an expectation-maximization algorithm. Accurate estimations of these conditional distributions enable us to predict probabilities of unlabeled data being ID or OOD. Moreover, we propose an alignment loss that further enhances the classification performance of this score. We combine these contributions with the self-supervision proposed in DoubleMatch and pseudo-labeling to form the Beta-model. Our proposed method demonstrates state-of-the-art results on many benchmark problems.
Contributions: Erik Wallin did the main work. Lennart Svensson, Fredrik Kahl, and Lars Hammarstrand supervised.

CHAPTER 5

Concluding Remarks and Future Work

Semi-supervised learning is an important domain of deep learning for enabling utilization of unlabeleled data, without the need for excessive human labeling work. Despite active research, determining optimal strategies to effectively use labeled and unlabeled data together remains an open question. Semi-supervised learning is often studied in the closed-set setting, where unlabeled data only contain known classes. For this setting, we proposed the method DoubleMatch in Paper A. DoubleMatch aims to improve methods relying on confidence-based pseudo-labeling. These methods only utilize unlabeled data with class predictions exceeding a confidence threshold. Other data are disregarded from the training loss. DoubleMatch addresses this limitation by combining the confidence-based pseudo-labeling with a self-supervised loss, which involves aligning predicted features of unlabeled data across different augmentations. The experimental evaluations in Paper A show that DoubleMatch improves training speed and final accuracy for many benchmark datasets. However, DoubleMatch does not outperform previous methods on all benchmark datasets and performs particularly poor in the low-label regime. For such scenarios, methods focusing on refined pseudo-labeling display more promising results [10], [14].

Open-set semi-supervised learning represents a more realistic and challenging setting, where uncurated unlabeled training sets may contain unknown classes, not seen in the labeled training set. In response to this scenario, we introduced SeFOSS in Paper B. SeFOSS follows the philosophy of DoubleMatch in that it aims to enable learning from all unlabeled data, in this case both ID and OOD data. To this end, SeFOSS incorporates the same form of self-supervised feature alignment as proposed by DoubleMatch. Moreover, it uses an energy-based method for determining if data are ID or OOD. Experimental results presented in Paper B show the robustness and superior overall performance across many open-set scenarios, when compared to existing methods for OSSL. In contrast to many previous works, SeFOSS places significant emphasis on OOD detection at test time, considering that unknown classes present in training data may also appear during testing. Moreover, an important observation from Paper B is that traditional methods for closed-set semi-supervised learning often outperform methods for open-set semi-supervised learning, even in the open-set scenario, when considering closed-set accuracy.

In Paper C we looked further into the problem of open-set semi-supervised learning. Paper C proposes a novel score for classification of ID and OOD that considerably increases performance on multiple benchmark problems. Furthermore, Paper C proposes the use of the expectation-maximization algorithm in OSSL for estimating conditional distributions of scores given ID and OOD data. With these estimated distributions, we can produce probabilistic predictions of data being ID or OOD.

For future research, an interesting research goal is to close the performance gap in terms of closed-set accuracy between closed-set semi-supervised learning and open-set semi-supervised learning. In theory, achieving accuracies comparable to the curated case with uncurated data should be possible, assuming we have access to equally many ID data. Potentially, it may even be possible to surpass the accuracy of the curated case, given that the OOD data contain information that can help us classify ID classes. Nevertheless, with current methods, some level of performance loss can generally be expected when comparing the curated case with the uncurated setting.

Additionally, we see a need to explore semi-supervised learning for domains beyond computer vision. Specifically for our research project, investigating semi-supervised learning for classification in radar is a notable goal. As many methods for semi-supervised learning are relying on domain-specific data aug-

mentation, transitioning from computer vision to radar introduces the challenge of identifying suitable augmentation strategies for the new domain. Furthermore, there is a significant disparity in the availability of public datasets for benchmarking between computer vision and a domain like radar. Addressing this disparity by contributing radar data is a goal of this research project.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[2] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, 2015.

[3] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, 2020.

[4] T. Brown, B. Mann, N. Ryder, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, 2020.

[5] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, 2020.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE conference on computer vision and pattern recognition*, 2009.

[7] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ questions for machine comprehension of text," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016.

[8] H. Pham, Z. Dai, Q. Xie, and Q. V. Le, "Meta pseudo labels," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[9] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009.

[10] Y. Wang, H. Chen, Q. Heng, *et al.*, "Freematch: Self-adaptive thresholding for semi-supervised learning," in *International Conference on Learning Representations*, 2023.

[11] Q. Xie, Z. Dai, E. Hovy, T. Luong, and Q. Le, "Unsupervised data augmentation for consistency training," in *Advances in Neural Information Processing Systems*, 2020.

[12] K. Sohn, D. Berthelot, N. Carlini, *et al.*, "FixMatch: Simplifying semi-supervised learning with consistency and confidence," in *Advances in Neural Information Processing Systems*, 2020.

[13] Y. Xu, L. Shang, J. Ye, *et al.*, "Dash: Semi-supervised learning with dynamic thresholding," in *International Conference on Machine Learning*, 2021.

[14] B. Zhang, Y. Wang, W. Hou, *et al.*, "FlexMatch: Boosting semi-supervised learning with curriculum pseudo labeling," in *Advances in Neural Information Processing Systems*, 2021.

[15] X. Zhai, A. Oliver, A. Kolesnikov, and L. Beyer, "S4L: Self-supervised semi-supervised learning," in *IEEE/CVF International Conference on Computer Vision*, 2019.

[16] I. Nassar, M. Hayat, E. Abbasnejad, H. Rezatofighi, and G. Haffari, "Protocon: Pseudo-label refinement via online clustering and prototypical consistency for efficient semi-supervised learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[17] M. Zheng, S. You, L. Huang, F. Wang, C. Qian, and C. Xu, "SimMatch: Semi-supervised learning with similarity matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[18] K. Saito, D. Kim, and K. Saenko, "Openmatch: Open-set semi-supervised learning with open-set consistency regularization," in *Advances in Neural Information Processing Systems*, 2021.

[19] S. Mo, J.-C. Su, C.-Y. Ma, *et al.*, "Ropaws: Robust semi-supervised representation learning from uncurated data," in *International Conference on Learning Representations*, 2023.

[20] Y. Wang, P. Qiao, C. Liu, G. Song, X. Zheng, and J. Chen, "Out-of-distributed semantic pruning for robust semi-supervised learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[21] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *Journal of Big Data*, 2019.

[22] L.-Z. Guo and Y.-F. Li, "Class-imbalanced semi-supervised learning with adaptive thresholding," in *International Conference on Machine Learning*, 2022.

[23] J. Kim, Y. Hur, S. Park, E. Yang, S. J. Hwang, and J. Shin, "Distribution aligning refinery of pseudo-label for imbalanced semi-supervised learning," *Advances in neural information processing systems*, 2020.

[24] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*. The MIT Press, 2006.

[25] H. Scudder, "Probability of error of some adaptive pattern-recognition machines," *IEEE Transactions on Information Theory*, vol. 11, no. 3, pp. 363–371, 1965.

[26] S. Fralick, "Learning to recognize patterns without a teacher," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 57–64, 1967.

[27] A. Agrawala, "Learning with a probabilistic teacher," *IEEE Transactions on Information Theory*, vol. 16, no. 4, pp. 373–379, 1970.

[28] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *33rd annual meeting of the association for computational linguistics*, 1995.

[29] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using em," *Machine learning*, vol. 39, pp. 103–134, 2000.

[30] D.-H. Lee *et al.*, "Pseudo-Label: The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on challenges in representation learning, ICML*, 2013.

[31] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko, "Semi-supervised learning with ladder networks," in *Advances in Neural Information Processing Systems*, 2015.

[32] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," in *International Conference on Learning Representations*, 2017.

[33] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[34] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems*, 2017.

[35] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 8, pp. 1979–1993, 2018.

[36] D. Berthelot, N. Carlini, E. D. Cubuk, *et al.*, "ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring," in *International Conference on Learning Representations*, 2020.

[37] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "RandAugment: Practical automated data augmentation with a reduced search space," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020.

[38] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016.

[39] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[40] F. Yang, K. Wu, S. Zhang, *et al.*, "Class-aware contrastive semi-supervised learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[41] J. Wang, T. Lukasiewicz, D. Massiceti, X. Hu, V. Pavlovic, and A. Neophytou, "Np-match: When neural processes meet semi-supervised learning," in *International Conference on Machine Learning*, 2022.

[42] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations*, 2018.

[43] V. Verma, A. Lamb, J. Kannala, Y. Bengio, and D. Lopez-Paz, "Interpolation consistency training for semi-supervised learning," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, 2019.

[44] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "MixMatch: A holistic approach to semi-supervised learning," in *Advances in Neural Information Processing Systems*, 2019.

[45] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[46] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, 2020.

[47] M. Caron, H. Touvron, I. Misra, *et al.*, "Emerging properties in self-supervised vision transformers," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[48] J.-B. Grill, F. Strub, F. Altché, *et al.*, "Bootstrap your own latent - a new approach to self-supervised learning," in *Advances in Neural Information Processing Systems*, 2020.

[49] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, 2019.

[50] H. Bao, L. Dong, S. Piao, and F. Wei, "Beit: Bert pre-training of image transformers," *International Conference on Learning Representations*, 2022.

[51] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *Proceedings of the IEEE international conference on computer vision*, 2015.

[52] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *International Conference on Learning Representations*, 2018.

[53] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European Conference on Computer Vision*, 2016.

[54] X. Chen and K. He, "Exploring simple siamese representation learning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[55] X. Wang, D. Kihara, J. Luo, and G.-J. Qi, "EnAET: A self-trained framework for semi-supervised and supervised learning with ensemble transformations," *IEEE Transactions on Image Processing*, vol. 30, pp. 1639–1647, 2020.

[56] E. Wallin, L. Svensson, F. Kahl, and L. Hammarstrand, "DoubleMatch: Improving semi-supervised learning with self-supervision," in *International Conference on Pattern Recognition*, 2022.

[57] Y. Chen, X. Zhu, W. Li, and S. Gong, "Semi-supervised learning under class distribution mismatch," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[58] L. Han, H.-J. Ye, and D.-C. Zhan, "On pseudo-labeling for class-mismatch semi-supervised learning," *Transactions on Machine Learning Research*, 2022.

[59] Z. Huang, J. Yang, and C. Gong, "They are not completely useless: Towards recycling transferable unlabeled data for class-mismatched semi-supervised learning," *IEEE Transactions on Multimedia*, 2022.

[60] Q. Yu, D. Ikami, G. Irie, and K. Aizawa, "Multi-task curriculum framework for open-set semi-supervised learning," in *European Conference on Computer Vision*, 2020.

[61] E. Wallin, L. Svensson, F. Kahl, and L. Hammarstrand, "Improving open-set semi-supervised learning with self-supervision," *arXiv preprint arXiv:2301.10127*, 2023.

[62] W. Liu, X. Wang, J. Owens, and Y. Li, "Energy-based out-of-distribution detection," in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2020.

[63] R. He, Z. Han, X. Lu, and Y. Yin, "Safe-student for safe deep semi-supervised learning with unseen-class unlabeled data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[64] H. Luo, H. Cheng, F. Meng, *et al.*, "An empirical study and analysis on open-set semi-supervised learning," *arXiv preprint arXiv:2101.08237*, 2021.

[65] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1501–1510.

[66] L.-Z. Guo, Z.-Y. Zhang, Y. Jiang, Y.-F. Li, and Z.-H. Zhou, "Safe deep semi-supervised learning for unseen-class unlabeled data," in *International Conference on Machine Learning*, 2020.

[67] X. Zhao, K. Krishnateja, R. Iyer, and F. Chen, "How out-of-distribution data hurts semi-supervised learning," in *2022 IEEE International Conference on Data Mining (ICDM)*, 2022.

[68] R. He, Z. Han, Y. Yang, and Y. Yin, "Not all parameters should be treated equally: Deep safe semi-supervised learning under class distribution mismatch," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

[69] Z. Huang, M.-J. Sidhom, B. Wessler, and M. C. Hughes, "Fix-a-step: Semi-supervised learning from uncurated unlabeled data," in *International Conference on Artificial Intelligence and Statistics*, 2023.

[70] J. Huang, C. Fang, W. Chen, *et al.*, "Trash to treasure: Harvesting ood data with cross-modal matching for open-set semi-supervised learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[71] J. Park, S. Yun, J. Jeong, and J. Shin, "Opencos: Contrastive semi-supervised learning for handling open-set unlabeled data," in *European Conference on Computer Vision*, 2022.

[72] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proceedings of the European conference on computer vision (ECCV)*, 2018.

[73] L.-Z. Guo, Y.-G. Zhang, Z.-F. Wu, J.-J. Shao, and Y.-F. Li, "Robust semi-supervised learning when not all classes have labels," *Advances in Neural Information Processing Systems*, 2022.

[74] K. Cao, M. Brbic, and J. Leskovec, "Open-world semi-supervised learning," in *International Conference on Learning Representations*, 2022.

[75] M. N. Rizve, N. Kardan, and M. Shah, "Towards realistic semi-supervised learning," in *European Conference on Computer Vision*, 2022.

[76] M. N. Rizve, N. Kardan, S. Khan, F. Shahbaz Khan, and M. Shah, "Openldn: Learning to discover novel classes for open-world semi-supervised learning," in *European Conference on Computer Vision*, 2022.

[77] J. Liu, Y. Wang, T. Zhang, Y. Fan, Q. Yang, and J. Shao, "Open-world semi-supervised novel class discovery," in *International Joint Conference on Artificial Intelligence*, 2023.

[78] P. Panareda Busto and J. Gall, "Open set domain adaptation," in *Proceedings of the IEEE international conference on computer vision*, 2017.

[79] K. Saito, S. Yamamoto, Y. Ushiku, and T. Harada, "Open set domain adaptation by backpropagation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018.

[80] J. Yang, H. Wang, L. Feng, *et al.*, "Semantically coherent out-of-distribution detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[81] F. Lu, K. Zhu, W. Zhai, K. Zheng, and Y. Cao, "Uncertainty-aware optimal transport for semantically coherent out-of-distribution detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.