Chapter 4
Learning with Limited Labelled Data

Christoffer Loeffler[1,2], Rasmus Hvingelby[2], Jann Goschenhofer[2]

Abstract Modern machine and deep learning require large amounts of training data. Yet, even if the data itself is abundantly available, the fraction of annotated data may still be proportionally small or missing. Hence, learning with limited labeled data is an important research field. Two streams of research attack this problem from opposite directions [64]. On the one hand, semi-supervised learning aims to leverage all information by directly incorporating unlabeled data. On the other hand, active learning finds unlabeled data for that annotations would be most beneficial for learning, and queries humans-in-the-loop of model training. This chapter discusses both concepts and their essential principles, methodological overlaps, and strengths and weaknesses. Furthermore, we elaborate on possible combinations and their advantages ands disadvantages. Finally, the conclusion refers to recent state-of-the-art and provides an outlook into the future of learning with few labeled data.

Key words: semi-supervised learning, active learning

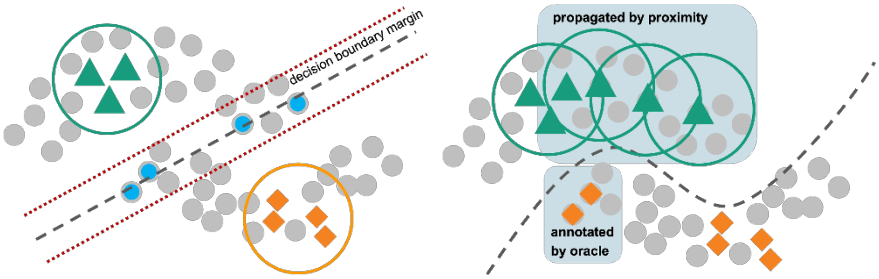
4.1 Introduction

One main hurdle in the design and training of machine learning models is their need for large amounts of labeled training data. This labeling process also referred to as the annotation process, can be very time consuming as it requires the knowledge and involvement of domain experts that add annotated input data X with their respective labels Y. Despite this abundance of labeled training data, there often exists a large amount of unlabeled data that was (not yet) annotated by domain experts. Due to

[1]Escuela de Ingeniería Informática, Pontificia Universidad Católica de Valparaíso, Valparaíso, Chile
[2]Fraunhofer Institute for Integrated Circuits IIS, Fraunhofer IIS, Nuremberg, Germany

Corresponding author: Christoffer Löffler
e-mail: christoffer.loffler@pucv.cl

C. Mutschler et al. (eds.), *Unlocking Artificial Intelligence*,
https://doi.org/10.1007/978-3-031-64832-8_4



(a) Intial imprecise decision boundary. (b) Exact decision boundary.

Fig. 4.1: In limited labeled data scenarios, only a subset of samples is annotated (orange and green data points) while the majority is unlabeled (grey data points). (a) shows the resulting imprecise and wrong decision boundary of an exemplary linear binary classifier trained on the labeled data only. As shown in (b), semi-supervised learning assumes that unlabeled samples are of the same class if they are close in proximity in the feature space. Semi-supervised methods such as label propagation, see upper part of (b), exploit this proximity to propagate the class information of the labeled samples. On the other hand, active learning aims to improve the learned model by selecting the most informative samples to annotate. Here, the annotator is queried with the most uncertain samples, i.e., those that are close to the decision boundary as depicted in the lower part of (b).

this, machine learning experts often face the situation of "learning with limited labeled data" where a small dataset of labeled data exists next to a large dataset of unlabeled data. Both semi-supervised and active learning try to leverage the information given in the unlabeled dataset next to the labeled dataset to train strong-performing machine learning models. Thereby, semi-supervised learning focuses on the direct incorporation of unlabeled data in the training process. Active learning on the other hand aims at finding those unlabeled samples that would support model training the most and presents them to a (human) oracle that iteratively annotates subsets of the unlabeled dataset in a model-driven way. Both approaches thereby "attack the same problem from opposite directions" [64] as illustrated in Figure 4.1. While semi-supervised methods exploit what the model thinks it knows about the unlabeled data, active methods attempt to explore the unknown aspects.

In the following, we provide an overview of both approaches and discuss their methodological overlaps, strengths, and weaknesses to give the reader a comprehensive understanding of both fields.

Throughout the chapter, we make use of the following notation. We define an input space $\mathcal{X}$ and use $y^{(i)} \in \mathcal{Y}$ to denote a categorical variable in the target space $\mathcal{Y}$ with a cardinality of $K = |\mathcal{Y}|$. Further, we define a labeled dataset $\mathcal{D}^l$ consisting of n_l tuples of samples and their respective labels $(x_i, y_i), ..., (x_l, y_l) \in \mathcal{D}^l$ as well as an unlabeled dataset $\mathcal{D}^u$ which consists of n_u samples $x_{l+1}, ..., x_u \in \mathcal{D}^u$. The goal of semi-supervised learning is to train a prediction model $f : \mathcal{X} \mapsto \mathcal{Y}$ on a

dataset $\mathcal{D} = (\mathcal{D}^l, \mathcal{D}^u)$ which consists of an labeled dataset $\mathcal{D}^l = \{(x^{(i)}, y^{(i)})\}_{i=1}^{n_l}$ and an unlabeled dataset $\mathcal{D}^u = \{x^{(i)}\}_{i=n_l+1}^{n}$ where $n = n_l + n_u$. Model predictions are denoted as $\hat{y}^{(i)} = f(x^{(i)}|\theta)$ where $\hat{y}^{(i)}$ is a class probability vector of dimension K, $\hat{y}^{(ik)}$ denotes the predicted probability for class $k \in 0, ..., K$ for input sample $x^{(i)}$ and θ refers to the model parameters. We consider the case where $n_l \ll n_u$, as usual in SSL. Further, we define one batch of data as $\mathcal{B} \subset \mathcal{D}$, where $\mathcal{B}^l \subseteq \mathcal{D}^l$ contains the labeled samples and $\mathcal{B}^u \subseteq \mathcal{D}^u$ the unlabeled samples in that batch such that $\mathcal{B} = (\mathcal{B}^l, \mathcal{B}^u)$.

4.2 Semi-Supervised Learning

The goal and promise of semi-supervised learning, at the intersection of unsupervised and supervised learning, is to leverage both labeled and unlabeled data for machine learning tasks. The expanding research in this field is mainly driven by the sometimes prohibitively high effort involved in annotating large labeled datasets on the one side and the abundance of unlabeled data on the other. Hence, semi-supervised methods mainly focus on settings with few labeled and many unlabeled training data. While there exists research on semi-supervised learning for a broad variety of learning tasks, we focus on semi-supervised classification for which most research exists.

Semi-supervised learning relies on three interconnected assumptions [72].

1. **Smoothness assumption**: two samples x_i, x_j that are close to each other in a high-density region of the input space should have similar labels y_i, y_j.
2. **Low-density assumption**: the decision boundary of model f should go through low-density areas where $p(x)$ is low, so-called low-density regions. This adds another perspective to assumption 1) as placing the decision boundary in a high-density region would violate this smoothness assumption.
3. **Manifold assumption**: high-dimensional data should lie on lower-dimensional manifolds, subspaces that are locally Euclidean. Hence, two samples x_i, x_j that lie on the same manifold are assumed to have similar labels y_i, y_j. This assumption mainly targets the curse of dimensionality and allows for the translation of the previous assumptions to high-dimensional settings.

Furthermore, semi-supervised algorithms can be distinguished in **inductive** and **transductive** methods. Inductive learning algorithms aim at learning a mapping $f : X \rightarrow Y$ from the data to the input space. After the learning phase at inference time, these models along their estimated model parameters can be used to assign predicted labels from Y to newly, unseen data X. Contrary to this, transductive methods merely aim at annotating the unlabeled D^u using the D^l such that $f : X^u \rightarrow Y$ without the learning of a general decision rule. In that sense, induction is more general as it aims at learning general decision rules while transduction tries to reason from the labeled cases to the specific unlabeled cases.

Following this introduction, we next provide an overview of classical semi-supervised learning methods and then focus on recent developments in deep semi-supervised learning and the different concepts applied therein.

4.2.1 Classical Semi-Supervised Learning

This section gives a rough overview of classical machine learning approaches developed for semi-supervised classification. Following the taxonomy developed in the standard textbook [13], we distinguish these models into four model classes: 1) **Generative models** such as the EM-algorithm for incomplete data [18] that aim at learning the class-conditional density $p(x|y)$ and use the unlabeled data D^u to improve its estimation. 2) Approaches that follow the **Low-Density Separation** rationale try to direct the decision boundary through low-density areas following the low-density assumption using the latent information in D^u to identify these areas. This mainly involves max-margin estimators such as the transductive SVM [17]. 3) **Graph-based methods** that exploit the neighborhood of labeled and unlabeled samples defined via a metric (e.g. defined via a kernel function following the manifold assumption). These neighborhood relationships are then used to propagate class labels from the labeled to their neighboring unlabeled samples. Most of these methods are transductive and Label Propagation [78] is one prominent method in this model class. 4) **Change of Representation**: two-step approaches that e.g. use D^u in the first step to learn a meaningful data representation which is then tailored towards the learning task using D^l in the second step.

4.2.2 Deep Semi-Supervised Learning

In a more recent overview, [72] extended this taxonomy further towards the use of neural networks along the dimensions of transduction and induction. Under transduction, they collect mainly Graph-based models that leverage joint neighborhood structures in $D = (D^l, D^u)$. With that, they follow the structure of Chapelle et al. [13] but extend it towards deep graph-based methods such as Deep Label Propagation [32].

They further differentiate different learning paradigms that mainly aim at extending existing supervised inductive methods, toward using additional unlabeled data D^u next to the labeled data D^l.

1) **Self-training** methods, also referred to as "Wrapper methods" or "Pseudo-Labeling", use a supervised model f trained on D^l to iterative pseudo-label additional unlabeled samples from D^u to augment the training data set and then re-train on this expanded D^{l*}.

2) **Unsupervised preprocessing** methods that use D^u to aid the generation of a meaningful representation of the data in an unsupervised manner. This includes

the extraction of meaningful features from the raw data to find an embedding that is favorable for the initial learning task. Such approaches contain but are not limited to dimensionality reduction techniques such as PCA or autoencoders, again related to the manifold assumption. Further, cluster-then-label approaches use clustering techniques over D or D^u only to facilitate the initial supervised learning task. The final sub-branch of methods mainly targeted at neural-network-based methods summarizes pre-training algorithms that use D^u to initialize the model architecture which is then fine-tuned on D^l.

3) **Inherently semi-supervised approaches** that extend supervised loss functions defined over D^l with tailored loss functions that allow the inclusion of D^u in the training process to enable a semi-supervised model training.

Recent strong-performing semi-supervised learning methods follow at least one of these paradigms or are combinations of them. In the remainder of this chapter, we will focus on 1) Self-training and 3) intrinsically semi-supervised learning as these are the most active research areas at the time of writing.

4.2.2.1 Self-training

Self-training, also referred to as Pseudo-labeling or Self-learning, is one of the oldest approaches to semi-supervised learning [62, 22, 2]. It follows the idea that the model trains itself by iteratively annotating parts of the unlabeled data. The procedure usually alternates between a training and a *pseudo-labeling* step. After the training step, the model selects unlabeled samples via a selection criterion such as model confidence. These selected samples are then assigned the predicted label and added from D^u to the now updated labeled dataset D^{l*}. The model is then trained on this (pseudo-) labeled dataset and this self-training cycle continues until a stopping criterion, such as the fact that no unlabeled data is left, is reached. This concept relates to Active Learning replacing the there often-used (human) oracle with the model f.

Self-training was transferred to deep learning by [40] and since then has sparked the creation of numerous variants. For instance, [3] investigate the confirmation bias that can occur when the model is overconfident but wrong on unlabeled samples D^u. This then leads to wrong pseudo-labels which confuses model training. They use Mixup [77] and the injection of label noise to overcome this issue. In a similar realm, [60] successfully use a combination of prediction confidence and model uncertainty with two distinct thresholds as a pseudo-label selection criterion to overcome this issue. [11] take in another perspective and combine *Curriculum Learning* with Pseudo-Labeling. This enables the model to use adaptive thresholds in the selection criterion and leads to on-par performance with more advanced and complex semi-supervised learning techniques. Next to these extensions, pseudo-labeling remains a crucial component in recent semi-supervised models.

4.2.2.2 Unsupervised Regularization

Alternative inherently semi-supervised methods create additional loss functions L^u defined over D^u or D which are combined with the initial, supervised loss function L^l to allow joint model training over both datasets via the combined loss $L = L^l + \lambda L^u$, where hyperparameter λ controls the impact of D^u. This has a regularizing effect and has the benefit that samples from D^u can be inherently integrated into model training. One early approach in this context is **Minimum Entropy Regularization** (MER) [27] where the prediction entropy serves as unsupervised regularization term such that $L^u(f, x_i) = H(f(x_i))$ for $x_i \in D^u$ leading to the final loss function

$$L(f, B^l, B^u) = -\sum_{(x^{(i)}, y^{(i)}) \in \mathcal{B}^l} \sum_{k=1}^{K} y^{(ik)} \log(\hat{y}^{(ik)}) - \lambda \sum_{(x^{(i)}) \in \mathcal{B}^u} \sum_{k=1}^{K} \hat{y}^{(ik)} \log(\hat{y}^{(ik)}) \tag{4.1}$$

This forces the model to create low entropy predictions, i.e., sharp predictions, over the entire dataset. MER was developed following the observation that unlabeled data does not contribute to the maximum-likelihood estimation of discriminative, supervised models. Thus, it introduces the regularization term as a prior adding an inductive bias to the model driven by the unlabeled data. The penalization of the model for high-entropy predictions over the unlabeled data potentially pushes the model's decision boundary towards low-density regions, following the *low-density assumption*. Originally developed for logistic regression, MER can also be used for neural-network-based classifiers.

The rationale of unsupervised regularization was further extended within models that use **Consistency Regularization**, also termed *perturbation-based methods*. These build up on the *smoothness assumption* such that a slightly perturbed version $\tilde{x}^{(i)} = x^{(i)}$ of the input sample $x^{(i)}$ is expected to have the similar class the clean, non-perturbed version $x^{(i)}$, assuming $x^{(i)}$ lies in a high-density region. This expected *consistency* in model predictions lends this branch of research its name. In recent years, different perturbation methods have been developed from the simple addition of random noise to inputs to the use of more elaborate methods which we will cover in the following.

Noise Perturbation. With the Ladder-Net, [55] introduced an Autoencoder-based approach that injects additive gaussian noise at different intermediate representations of the input samples and calculates a regularization term over changes in these representations. This allows them to a) robustify the model representations and b) train the model on the joint D using both the reconstruction loss of the autoencoder as well as the noise-regularization term. The encoder part of the architecture is used at inference time. Instead of random noise, [51] propose to add directed *adversarial noise* to the unlabeled input samples as a regularization mechanism. In contrast to the addition of noise to the input sample, the Π-Model adds noise in the form of dropout layers to the model architecture. The regularization term is then calculated over different model prediction samples via the MCDropout algorithm [23] which

simulates an ensemble of models and enforces consistent model predictions across those.

Temporal Consistency. Another branch of research leverages predictions from different training stages as a perturbation mechanism following the rationale that the model should produce *temporally consistent* model predictions during training. Within the Temporal Ensembling Model, [38] maintain an exponential moving average of model predictions over stochastically augmented, unlabeled input samples from past training epochs. In the current training epoch, these serve as an auxiliary target and the squared distance between those past model predictions is used as an unsupervised loss function L^u. [70] follow this rationale as well in their Mean Teacher architecture. Instead of storing past model predictions of D^u, they maintain a teacher version of the initial student model whose weight parameters are updated via exponential moving averaging of the student model's weights that are directly optimized via gradient descent. Model predictions over D^u from the teacher model here serve as auxiliary targets in the unsupervised loss part L^u. This concept remains an important training paradigm for semi-supervised learning and was used in the Unbiased Teacher architecture for semi-supervised object detection [44].

4.2.3 Self-Training and Consistency Regularization

The use of elaborate data augmentation strategies as perturbation methods in consistency regularization sparked a more recent line of research in this area. Within MixMatch [9], the authors combine a) data augmentation with the different established semi-supervised techniques b) Pseudo-Labeling and Entropy Regularization via a Sharpening function, and c) Mixup [77] in one holistic approach to semi-supervised learning. Model prediction vectors over differently augmented versions of an unlabeled sample $x_u^{(i)}$) are averaged, sharpened via a temperature scaling mechanism, and then used as pseudo-labels. Subsequently, a batch of labeled and pseudo-labeled data are combined via MixUp to create synthetic training samples with are then fed into a Brier-Score as an unsupervised loss function L^u. This combination of different semi-supervised learning paradigms allows MixMatch to yield impressive predictive performance given low levels of supervision. With FixMatch, [68] improve upon these results by introducing the strong- and weak- augmentation scheme: pseudo-labels from weakly augmented samples $x_u^{(i)}$ are selected based on a prediction confidence criterion and serve as training targets in the auxiliary loss L^u. Model predictions over exaggeratedly strong augmented versions of these samples are then used as input to this loss function, allowing model training on both D^u and D^l. This idea has sparked a lot of further research such as FeatMatch [37] which uses data augmentation in the manifold space or FlexMatch [76] which combines this concept with Curriculum Learning.

4.3 Active Learning

Active Learning (AL) algorithms select the most valuable samples and query an annotator with them [64]. This means that models can learn more quickly from a subset of annotated samples and that the intelligent choice of such samples can be better than randomly subsampling a data stream or dataset. This choice can be based on, e.g., insights about the model, the dataset, or on a heuristic.

Cost. The reduction of cost is one of the primary reasons to use Active Learning. The costs arise from different sources, e.g., the annotation task's difficulty and the associated expensive expertise of the annotators. Similarly, creating well-curated, representative datasets may become a financial roadblock for ML projects. A typical example of these issues is the medical field [7], where Active Learning may decrease the time (and money) spent on generating labels.

Active Learning Loop. The human-in-the-loop, that is also called the "oracle", is at the center of the AL loop [64]. Figure 4.2 shows the loops components: the ML model, a pool of labeled training data, and a pool of unlabeled data from which the Active Learner constructs queries to the annotator. A common assumption is that the oracle initially annotates a small subset for a first model training. Next, an AL strategy selects one or more unlabeled samples using an acquisition function. The expectation is that the ML model would learn faster from these than from randomly sampled data. Finally, the expert is queried and the labeled data is added to the labeled pool. The stop condition may be the depletion of some budget or an accuracy threshold.

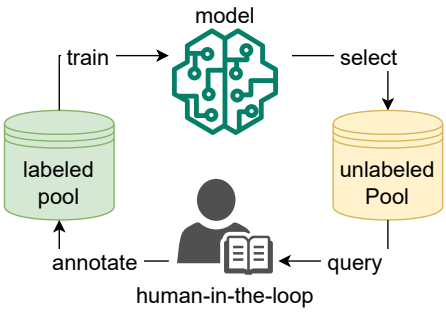


Fig. 4.2: The Active Learning loop has a human-in-the-loop at its center, that serves as the label oracle. The AL method acquires samples to query the oracle with, and the subsequent labels are added to a training pool.

Scenarios. The literature distinguishes how the unlabeled data is made available [64]. In pool-based Active Learning, all unlabeled data is available in a database, and any of these can be selected for the construction of a query. Alternatively, in stream-based scenarios samples may stream into the AL loop over time, and the method incrementally selects queries. Besides the delivery mode of the dataset, the number of simultaneously selected samples further differentiates what acquisition functions can achieve and how queries can be selected. Querying the oracle one sam-

ple at a time may choose a best sample but requires frequent re-fitting of the model, whereas batch-mode AL samples whole batches at once and thus speeds up the AL loop considerably. This is especially important when using DNNs as models [4].

Acquisition Functions. The method of how the AL algorithm chooses which unlabeled samples an oracle is queried with is a crucial part of the AL loop. A typical example of an acquisition functions for classification tasks is querying the least confident prediction [41]. These are $\hat{x} = \text{argmax}_x(1 - P(\hat{y}|x))$, where $\hat{y} = \text{argmax}_y(P(y|x))$ is the most likely label $\hat{y}$ for an unlabeled sample x. This uncertainty-based acquisition function selects samples that the model is least certain about.

Before the recent success of DL, a multitude of acquisition functions was proposed [64], such as uncertainty-based sampling, queries by committees of models, based on the expected model change, the expected error reduction, on variance reduction or based on density. The following sections explain the fundamental concepts of Deep Active Learning, such as uncertainty and diversity sampling, and their combination.

4.3.1 Deep Active Learning (DAL)

Deep Neural Networks require large amounts of data to generalize well [58]. This becomes an issue in supervised learning settings that, unlike self-, semi- or unsupervised learning, need annotations to fit models. Using AL may seem the natural choice to reduce the costs for generating training datasets. However, DAL faces challenges that arise from their use of DNNs. Traditional heuristics like discussed in [64] and one-by-one querying showed to be ineffective when used with DNNs [63]. Hence, this section introduces the fundamentals of AL with a focus on models from Deep Learning. The research on DAL [58] developed families of methods that broadly parallel traditional AL strategies [64], but adapted them to DL. Hence, the following sections explain the more traditional AL strategies. See Section 4.5 for an outlook on more recent literature that extends the fundamentals.

4.3.2 Uncertainty Sampling

Selecting those samples for queries, that a model is least certain about [41], intuitively should provide most information on the dataset. For probabilistic models this is a feasible approach [64]. However, depending on the task, obtaining predictive uncertainty for DNNs is unavailable or of lower quality. In classification tasks, the softmax activation tends to quantify uncertainty with overconfidence, and regression usually is not accompanied by an uncertainty measure at all. Obtaining a more reliable uncertainty measure is important to select those samples, that are really informative.

DNNs predictive uncertainty can be interpreted as having two components: on the one hand of aleatoric or data uncertainty, and on the other of epistemic or model uncertainty [31]. The model uncertainty can be reduced via Active Learning by selecting samples for that the model is least certain. While approaches like Bayesian neural networks provide a well-calibrated uncertainty estimation that may be used for AL, they are intractable for larger problem instances.

Recently, Gal et al. [24] proposed a first tractable and efficient approach for estimating a DNN's uncertainty that is implemented as a Bayesian Convolutional Network. This approach provides better calibrated uncertainty by implementing an ensemble- or vote-agreement scheme [39] based on a Monte-Carlo simulation of a model ensemble using the Dropout connections of the DNN. This trick allows to interpret each MC pass as a separate model and thus the epistemic uncertainty is measured more efficiently.

Beluch et al. [7] proposed to use *the power of ensembles* for AL, and show that this source of uncertainty is better calibrated than relying on Dropout connections within a single network. However, this observation was only valid in the few data domain. Interestingly, they show that AL with an ensemble still leads to increases in accuracy in larger problem sizes.

These two uncertainty-based Active Learners provide measures of uncertainty that different acquisition functions use to select queries. We present the three most common functions. The first is based on Shannon's information theory [65]. The **Max Entropy** function selects those points, that maximize the predictive entropy as follows

$$H[y|xD^l] = -\sum (p(y = k)|x, D^l)\log(p(y = k|x, D^l) \tag{4.2}$$

This is then adapted for ensembles and MC Dropout, in that the probabilities $p(\cdot|\cdot)$ are summed and averaged over the number of networks in the ensemble [7] or over the number of forward passes [24].

The **Variation Ratio** acquisition function selects those samples, whose predicted classes have the lowest agreement in an ensemble, see Eq. 4.3, or in its Bayesian formulation, whose probability is more dispersed to others, see Eq. 4.4:

$$\text{variation-ratio}(x) = 1 - \frac{m}{N}, \tag{4.3}$$

$$= 1 - max_y p(y|x, D) \tag{4.4}$$

4.3.3 Diversity Sampling

Another parallel between classical and Deep Active Learning is the notion of querying the oracle with a diverse set of examples, so that the model learns from a representative training dataset. The selection of a diverse batch seems especially promising for batch-mode learning, because it can help avoiding biased training. Compared to a simple random down-sampling of the training data pool, AL strate-

gies, such as the core-set selection proposed by Sener and Sevarese [63] aim to find an optimal (unbiased) subset. Uncertainty-based sampling generally is also more affected by outliers [64]. Additionally, Sener and Savarese empirically show the limitations of uncertainty-based AL when used with larger datasets.

In traditional optimization, algorithms for selecting a core-set were already used for k-center clustering and other applications [1], and also already for AL, e.g., with Support Vector Machines [71]. However, the extension of this idea to a deep model, such as CNNs, was only recently pioneered [63]. The authors use the DNN to generate an embedding of the pool, and then solve the k-center problem to select a batch query. In addition, Yehuda et al. [74] recently proposed a diversity sampling approach that maximizes Probability Coverage, and that is designed specifically for the low-budget regime.

4.3.4 Balanced Criteria

The combination of multiple selection criteria lends itself especially well to DAL, because model training is most often performed via mini-batches and stochastic gradient descent, and the selection of samples within such a batch enables it. This section explains BALD [29] and BatchBALD [35] as examples for the variety of AL strategies that combine uncertainty-sampling with selecting more diverse batches.

Houlsby et al. [29] propose BALD, that measures the mutual information between the model's parameters and its predictions, which points out whether learning about the true label would provide new information on the parameters as well. BALD uses the following equation at its core:

$$I(y; w|x, D^l) = H(y|x, D^l) - E_{p(w|D^l)}[H(y|x, w, D^l)]. \tag{4.5}$$

The first term measures the prediction's entropy and is high for uncertain predictions. The second term is the expectation of the prediction, given the model and its parameters, and is low if the model is certain. Maximizing the information I leads to choosing samples with a high uncertainty in the prediction, but a low uncertainty in the learned model. However, this does not select for diverse samples and performs bad with larger batch sizes [35].

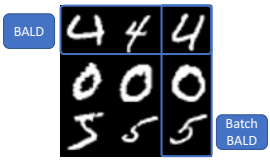


Fig. 4.3: The idealized acquisition of BatchBALD [35] selects a more diverse query compared to BALD [29], that selects the most informative samples, even if they repeat.

Kirsch et al. [35] propose BatchBALD as an extension to BALD, that finds batches of informative data. They extend Eq. 4.7 by estimating the joint of multiple data points $x_1, ..., x_b$:

$$\begin{aligned} I(y_1, ..., y_b; w|x_1, ..., x_b, D^l) =& H(y|x_{1:b}, D^l) \\ & - E_{p(w|D^l)}[H(y|x_{1:b}, w, D^l)]. \end{aligned} \tag{4.6}$$

Kirsch et al. argue that BALD overestimates the joint mutual information of pairs of prediction y_i and sample x_i, whereas the formulation of I for BatchBALD measures the overlap between multiple variables 1 to b from a batch, and thus tends to acquire more diverse queries. Figure 4.3 visualizes this with BALD ignoring the repetitiveness of similar x_i within a batch, whereas BatchBALD considers $x_{1:b}$ in calculating Entropy H.

4.4 Active Semi-Supervised Learning

Both AL and SSL aim to improve learning with limited labeled data. They tackle the problem from two different perspectives, where SSL assumes a static setting where the set of labeled data is fixed, AL assumes a dynamic setting where expanding the labeled data pool is possible. In the AL setting the model will be trained every time new labeled data is available, i.e., in between each of the query iterations. Most AL approaches only use the labeled data for training the model. However, in a pool-based AL setting, both labeled data and unlabeled data are available to the model. Thus it would be a natural idea to use SSL techniques that do not only use labeled data but also unlabeled data for training.

As AL and SSL are compatible, several works have combined them and we will look at how this can be done. First, we will give an example of how AL and SSL can be combined using concepts introduced in previous sections. Second, we will discuss the mutual benefits of SSL and AL based on recent advancements in SSL.

4.4.1 How can SSL and AL Work Together?

Following the AL loop depicted in Figure 4.2 a pool of unlabeled data and a pool of labeled data is available. This data will be used for training a model by minimizing an objective loss function $\mathcal{L}$. Commonly in AL the loss for training the model is only the supervised loss $\mathcal{L} = \mathcal{L}_{sup}$, which could be standard cross-entropy loss for classification. When integrating SSL into the AL loop we utilize the unlabeled data as well by combining the supervised loss with an unsupervised loss $\mathcal{L} = \mathcal{L}_{sup} + \mathcal{L}_{unsup}$. As seen previously, there exists a variety of different loss functions for the SSL loss which can be combined with any AL acquisition function.

Gao et al. [25] suggested to combine SSL and AL and used consistency regularization as basis for their SSL loss. They experimented with multiple acquisition functions such as random sampling, the uncertainty-based max-entropy and diversity-based k-center clustering. However, they note that choosing the right acquisition function is crucial when combining SSL and AL. As they are using consistency regularization as their unsupervised loss $\mathcal{L}_{unsup}$ that enforces consistency of predictions over augmented versions of the sample, they hypothesize that labeling the samples with the most inconsistent predictions should be the most beneficial to the model. This is based on the intuition that these samples must be hard to classify for the model. They propose the following simple metric for acquisition

$$\varepsilon(x^{(i)}) = \sum_{k}^{K} \mathrm{Var}\big[p(y^{(ik)} = k|x^{(i)}), p(y^{(ik)} = k|\tilde{x}_1^{(i)}), \ldots, p(y^{ik} = k|\tilde{x}_N^{(i)})\big], \tag{4.7}$$

where N is the number of different data augmentations applied.

Combining consistency regularization and a maximum inconsistency acquisition function [25] show that SSL benefits AL and outperforms other AL methods from the literature. They also show that their specific choice of SSL method and AL acquisition function performs better than other combinations.

4.4.2 Are SSL and AL Always Mutually Beneficial?

The recent advances in Deep Semi-Supervised Learning have shown impressive performance in utilizing the unlabeled data together with a small amount of labels [69]. This progress has raised the question of whether the human annotations are beneficial when SSL is able to utilize the unlabeled data so efficiently and therefore questioning the relevance of AL [50, 12, 8].

Utilizing consistency regularization-based SSL in the AL loop, [50] showed that for image classification the combination of SSL and random sampling works better than using AL for sampling. Similarly, [12] do not observe any additional benefit of using more advanced AL algorithms when combined with both SSL and self-supervised methods. [8] experimented with self-supervised models and active learning and demonstrated that self-supervised learning in itself is more efficient than AL at reducing the labeling effort. They also observe that the combination of self-supervised learning and AL is only beneficial only when the labeling budget is high which goes against the purpose of using AL.

Although these recent critiques of AL show the impressive performance of SSL, more research is needed to understand if this is also the case in real-world scenarios. The comparisons between AL and SSL [50, 12, 8] are based on experimental results on well-established benchmark datasets such as CIFAR10 and CIFAR100 where it is well-known what data augmentations and hyperparameters work well. This is not the case in real-world settings where it is hard to find optimal hyperparameters as well as design data augmentations that are label preserving and beneficial to the model.

As semi-supervised methods rely on assumptions it cannot be guaranteed that they will perform well in case these assumptions are broken. Therefore it is important to analyze a real-world scenario and conduct experiments to see if SSL is actually beneficial [72].

In this context, [53] formulated a critique on the evaluation of semi-supervised learning techniques. Among other issues, they observed that the multitude of hyperparameters such as weighting factors, thresholds, or perturbation ratios, require heavy hyperparameter tuning. This in turn requires the presence of reasonably large labeled validation data sets whose size is often magnitudes higher than that of the labeled training dataset – which increases the required amount of labeled training data for practical scenarios. While semi-supervised learning promises the effective use of unlabeled data for model training, its final benefit in practical scenarios heavily depends on the final setting – semi-supervised learning can help alleviate the problem of limited labeled data but is no silver bullet to it.

4.5 Conclusion and Outlook

Semi-supervised learning tries to tackle the limited labeled data problem by using the latent information provided in a large, unlabeled dataset D^u next to a smaller labeled dataset D^l. The field has been around from the early days of Machine Learning research and spans various approaches and related research fields. While recent deep semi-supervised learning approaches yield impressive gains on benchmark datasets, their applicability to practical real-world scenarios depends on the respective task and cannot be taken for granted. For instance, the heavy use of tailored data augmentation strategies in modern, strong-performing semi-supervised learning methods requires strong domain knowledge of the underlying task which could also be used to annotate more training data. These approaches also mainly target scenarios with a balanced class distribution and assume that both D^l and D^u follow the same data distribution, i.e. the absence of any distribution shifts. Further, the design and training of these partially highly elaborate algorithms require intense engineering and modeling efforts which could alternatively be used to annotate more high-quality training data.

Despite this critique, semi-supervised learning offers a high potential for low-label scenarios which fuels evermore increasing research activity in this field. Recent algorithmic development merges the concept of semi-supervision with connected research fields such as contrastive learning. With S4L, semi-supervised self-supervised learning, [75] introduced a self-supervised training scheme building up on contrastive learning for model training in a semi-supervised fashion. Furthermore, [14] successfully combined self-supervised pretraining on unlabeled data via the SimCLR architecture with subsequent semi-supervised fine-tuning and showed impressive classification performance on the ImageNet benchmark with a small 1% fraction of labeled examples. Similarly, [46] leverage self-supervised learning to extend FixMatch towards barely supervised learning scenarios, where as little as 4 labeled

samples per class are provided. With the advent of multi-purpose and multi-modal models [54], we can expect the use of these large pretrained models also for the generation of pseudo-labels in semi-supervised image classification tasks, similar to their use in Natural Language Processing.

In summary, Active Learning is a method to increase the number of annotated samples in the most cost-efficient manner [64]. Two of the fundamental strategies are sampling according to an uncertainty measure of the model [7, 24] or according to a representative measure of the underlying data distribution [63]. A combination [35] of such concepts balances the classical explore/exploit dilemma.

Most recent research on modern Deep Learning [59] can be broadly categorized as learning to active learning. For example, the underlying data distributions can be learned with generative models and this representation exploited [67, 48, 66, 34]. AL can be understood as an optimization problem and solved using Reinforcement Learning [73, 5, 36, 10, 19, 57, 45, 28], by imitating experts [61, 43, 47], or by simply selecting the most suitable strategy from a diverse set of heuristics [6, 30, 16] In closely related fields, AL can be cast as Meta-Learning of learning quickly with few samples [56, 15, 20, 52, 33, 21, 49, 42] or even as a Neural Architecture Search problem [26].

The future of Active Learning and Semi-Supervised Learning could be a combination that leverages both methods' strengths (active sampling) to reduce their weaknesses (cost, uncertainty). When combining the two paradigms, it is important that the chosen acquisition functions are designed to benefit an SSL method where the SSL method is unable to utilize the unlabeled data.

References

1. P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan. Geometric Approximation via Coresets. *Combinatorial and Computational Geometry*, MSRI Publications(52):21, 2005.
2. A. Agrawala. Learning with a probabilistic teacher. *IEEE Transactions on Information Theory*, 16(4):373–379, 1970.
3. E. Arazo, D. Ortego, P. Albert, N. E. O. Connor, and K. Mcguinness. Pseudo-Labeling and Confirmation Bias in Deep Semi-Supervised Learning. In *IJCNN*, 2020.
4. J. T. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal. Deep Batch Active Learning by Diverse, Uncertain Gradient Lower Bounds. 2019.
5. P. Bachman, A. Sordoni, and A. Trischler. Learning Algorithms for Active Learning. *Proceedings of the 34th International Conference on Machine Learning*, 2017.
6. Y. Baram, R. El-Yaniv, and K. Luz. Online choice of active learning algorithms. *Journal of Machine Learning Research*, 5:255–291, 2004.
7. W. H. Beluch, T. Genewein, A. Nürnberger, and J. M. Köhler. The Power of Ensembles for Active Learning in Image Classification. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 9368–9377, 2018.
8. J. Z. Bengar, J. van de Weijer, B. Twardowski, and B. Raducanu. Reducing label effort: Self-supervised meets active learning. *CoRR*, abs/2108.11458, 2021.
9. D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in Neural Information Processing Systems*, 32, 2019.

10. A. Casanova, P. O. Pinheiro, N. Rostamzadeh, and C. J. Pal. REINFORCED ACTIVE LEARNING FOR IMAGE SEGMENTATION. page 17, 2020.
11. P. Cascante-Bonilla, F. Tan, Y. Qi, and V. Ordonez. Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning. *AAAI*, 2021.
12. Y.-C. Chan, M. Li, and S. Oymak. On the Marginal Benefit of Active Learning: Does Self-Supervision Eat Its Cake? *arXiv:2011.08121 [cs]*, Nov. 2020.
13. O. Chapelle, B. Scholkopf, and A. Zien. Semi-supervised learning. *Cambridge, Massachusettes: The MIT Press View Article*, 20(3):542–542, 2009.
14. T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton. Big self-supervised models are strong semi-supervised learners. *NeurIPS*, 2020.
15. Y. Chen, M. W. Hoffman, S. G. Colmenarejo, M. Denil, T. P. Lillicrap, M. Botvinick, and N. De Freitas. Learning to learn without gradient descent by gradient descent. *34th International Conference on Machine Learning, ICML 2017*, 2:1252–1260, 2017.
16. H. M. Chu and H. T. Lin. Can active learning experience be transferred? *Proceedings - IEEE International Conference on Data Mining, ICDM*, pages 841–846, 2017.
17. R. Collobert, F. Sinz, J. Weston, L. Bottou, and T. Joachims. Large scale transductive svms. *Journal of Machine Learning Research*, 7(8), 2006.
18. A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *J. Royal Statistical Society: Series B*, 39(1):1–22, 1977.
19. Y. Fan, F. Tian, T. Qin, X.-Y. Li, and T.-Y. Liu. Learning to Teach. pages 1–16, 2018.
20. C. Finn, P. Abbeel, and S. Levine. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. 2017.
21. C. Finn, K. Xu, and S. Levine. Probabilistic Model-Agnostic Meta-Learning. In *NIPS*, number NeurIPS, 2018.
22. S. Fralick. Learning to recognize patterns without a teacher. *IEEE Transactions on Information Theory*, 13(1):57–64, 1967.
23. Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.
24. Y. Gal, R. Islam, and Z. Ghahramani. Deep Bayesian Active Learning with Image Data. Technical report, 2017.
25. M. Gao, Z. Zhang, G. Yu, S. O. Arik, L. S. Davis, and T. Pfister. Consistency-based Semi-supervised Active Learning: Towards Minimizing Labeling Cost. In *ECCV*, 2020.
26. Y. Geifman and R. El-Yaniv. Deep Active Learning with a Neural Architecture Search. In *Conference on Neural Information Processing Systems*, page 11, Vancouver, Canada, 2019.
27. Y. Grandvalet, Y. Bengio, et al. Semi-supervised learning by entropy minimization. *NeurIPS*, 367:281–296, 2005.
28. M. Haussmann, F. A. Hamprecht, and M. Kandemir. Deep Active Learning with Adaptive Acquisition. In *Intl. Joint Conf. on Artificial Intelligence*, pages 2470–2476, 2019.
29. N. Houlsby, F. Huszár, Z. Ghahramani, and M. Lengyel. Bayesian Active Learning for Classification and Preference Learning, Dec. 2011.
30. W. N. Hsu and H. T. Lin. Active learning by learning. *Proceedings of the National Conference on Artificial Intelligence*, 4:2659–2665, 2015.
31. E. Hüllermeier and W. Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. 110(3):457–506.
32. A. Iscen, G. Tolias, Y. Avrithis, and O. Chum. Label propagation for deep semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5070–5079, 2019.
33. M. A. Jamal and H. Cloud. Task Agnostic Meta-Learning for Few-Shot Learning. 2018.
34. K. Kim, D. Park, K. I. Kim, and S. Y. Chun. Task-Aware Variational Adversarial Active Learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8162–8171, Nashville, TN, USA, June 2021. IEEE.
35. A. Kirsch, J. van Amersfoort, and Y. Gal. BatchBALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning. (NeurIPS), 2019.

36. K. Konyushkova, R. Sznitman, and P. Fua. Learning Active Learning from Data. In *Proceedings of the 31st Conference n Neural Information Processing Systems*, 2017.
37. C.-W. Kuo, C.-Y. Ma, J.-B. Huang, and Z. Kira. Featmatch: Feature-based augmentationfor semi-supervised learning. In *European Conference on Computer Vision*. Springer, 2020.
38. S. Laine and T. Aila. Temporal ensembling for semi-supervised learning. *ICLR*, 2017.
39. B. Lakshminarayanan, A. Prinzel, and C. Blundell. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In *NeurIPS*, 2017.
40. D.-H. Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning*, volume 3, page 896, 2013.
41. D. D. Lewis, T. B. Laboratories, and M. Hill. A Sequential Algorithm for Training Text Classifiers. page 10, 1994.
42. M. Li, X. Liu, J. van de Weijer, and B. Raducanu. Learning to Rank for Active Learning: A Listwise Approach. *arXiv*, (i), 2020.
43. M. Liu, W. Buntine, and G. Haffari. Learning how to actively learn: A deep imitation learning approach. In *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, volume 1, pages 1874–1883. Association for Computational Linguistics, 2018.
44. Y.-C. Liu, C.-Y. Ma, Z. He, C.-W. Kuo, K. Chen, P. Zhang, B. Wu, Z. Kira, and P. Vajda. Unbiased teacher for semi-supervised object detection. *ICLR*, 2021.
45. Z. Liu, J. Wang, S. Gong, D. Tao, and H. Lu. Deep Reinforcement Active Learning for Human-in-the-Loop Person Re-Identification. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6121–6130, Seoul, Korea (South), Oct. 2019. IEEE.
46. T. Lucas, P. Weinzaepfel, and G. Rogez. Barely-supervised learning: semi-supervised learning with very few labeled images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1881–1889, 2022.
47. C. Löffler and C. Mutschler. Iale: Imitating active learner ensembles. *Journal of Machine Learning Research*, 23(107):1–29, 2022.
48. D. Mahapatra, B. Bozorgtabar, J.-P. Thiran, and M. Reyes. Efficient Active Learning for Image Classification and Segmentation Using a Sample Selection and Conditional Generative Adversarial Network. In A. F. Frangi, J. A. Schnabel, C. Davatzikos, C. Alberola-López, and G. Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, volume 11071, pages 580–588. Cham, 2018.
49. N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel. A Simple Neural Attentive Meta-Learner. In *ICLR*, pages 1–17, 2018.
50. S. Mittal, M. Tatarchenko, Ö. Çiçek, and T. Brox. Parting with illusions about deep active learning. *ArXiv*, abs/1912.05361, 2019.
51. 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*, 41(8):1979–1993, 2018.
52. A. Nichol, J. Achiam, and J. Schulman. On First-Order Meta-Learning Algorithms. pages 1–15, 2018.
53. A. Oliver, A. Odena, C. A. Raffel, E. D. Cubuk, and I. Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. *Advances in neural information processing systems*, 31, 2018.
54. A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763, 2021.
55. A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko. Semi-supervised learning with ladder networks. *Advances in neural information processing systems*, 28, 2015.
56. S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. *ICLR*, 2017.
57. S. Ravi and H. Larochelle. Meta-Learning for Batch mode Active Learning. *ICLR workshop*, pages 1–6, 2018.
58. P. Ren, Y. Xiao, X. Chang, P. Y. Huang, Z. Li, X. Chen, and X. Wang. A survey of deep active learning. *arXiv*, 37(4), 2020.

59. P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B. B. Gupta, X. Chen, and X. Wang. A Survey of Deep Active Learning. *ACM Comput. Surv.*, 54(9):1–40, Dec. 2022.
60. M. N. Rizve, K. Duarte, Y. S. Rawat, and M. Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *ICLR*, 2021.
61. S. Ross, G. J. Gordon, and J. A. Bagnell. A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. page 9, 2011.
62. H. Scudder. Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory*, 11(3):363–371, 1965.
63. O. Sener and S. Savarese. Active learning for convolutional neural networks: A core-set approach. In *Intl. Conf. Learning Representations*, Vancouver, CA, 2018.
64. B. Settles. Active Learning Literature Survey. Technical Report 1, Morgan & Claypool Publishers, 2012.
65. C. E. Shannon. A Mathematical Theory of Communication. *The Bell System Technical Journal*, (27):379–423, 1948.
66. C. Shui, F. Zhou, C. Gagne, and B. Wang. Deep Active Learning: Unified and Principled Method for Query and Training. In *Deep Active Learning*, page 10, Palermo, Italy, 2020.
67. S. Sinha, U. C. Berkeley, and T. Darrell. Variational Adversarial Active Learning. 2019.
68. K. Sohn, D. Berthelot, C. Li, Z. Zhang, N. Carlini, E. D. Cubuk, A. Kurakin, H. Zhang, and C. Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *CoRR*, abs/2001.07685, 2020.
69. K. Sohn, D. Berthelot, C.-l. L. Zizhao, Z. Nicholas, E. D. Cubuk, A. Kurakin, H. Zhang, and C. Raffel. FixMatch : Simplifying Semi-Supervised Learning with Consistency and Confidence. In *NeurIPS*, 2020.
70. A. Tarvainen and H. Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017.
71. I. W. Tsang, J. T. Kwok, P.-M. Cheung, C. U. Hk, C. U. Hk, and C. U. Hk. Core Vector Machines: Fast SVM Training on Very Large Data Sets. *Journal of Machine Learning Research*, 6(13):29, 2005.
72. J. E. Van Engelen and H. H. Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020.
73. M. Woodward and C. Finn. Active One-shot Learning. In *NIPS*, Barcelona, Spain, 2016.
74. O. Yehuda, A. Dekel, G. Hacohen, and D. Weinshall. Active learning through a covering lens. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 22354–22367, 2022.
75. X. Zhai, A. Oliver, A. Kolesnikov, and L. Beyer. S4l: Self-supervised semi-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1476–1485, 2019.
76. B. Zhang, Y. Wang, W. Hou, H. WU, J. Wang, M. Okumura, and T. Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 18408–18419, 2021.
77. H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. Mixup: Beyond empirical risk minimization. *ICLR*, 2018.
78. X. Zhul and Z. Ghahramanih. Learning from labeled and unlabeled data with label propagation. 2002.