Deep Active Learning in the Presence of Label Noise

Moseli Mots'oehli

Department of Information and Computer Science,
University of Hawai'i at Manoa
moselim@hawaii.edu

1 Introduction

Machine learning algorithms are a sub-class of artificial intelligence that learns from data to perform a pre-defined task such as classification, regression, or clustering. Of the numerous algorithms for machine learning, artificial neural networks, deep neural networks in particular have done exceptionally well in tasks involving complex data representations such as images, text, and sound. The main reason for this is that if you have a large enough dataset, you can build more extensive and more complex models with little to no risk of over-fitting. While this works in theory, the practical applications have major drawbacks such as the need for labeled training examples that come at a high cost due to the time needed to label the data, the high cost of labor in very specialized fields, or the cost of running simulations that would produce the ground truth dataset. The solution comes in the form of deep active learning (DAL) algorithms, which strive to let the learning algorithm iteratively pick data examples to be labeled from a larger unlabelled dataset, in such a manner that results in: (1) A smaller labeled training set, (2) A dataset that is representative of the underlying data distribution leading to a near-optimal learner, (3) A data labeling skim that does not exceed the labeling budget.

While this works well for most use cases, real-world dataset labeling has inherent label noise due to a variety of factors such as redundant observations being labeled differently, the best human expert classification performance being low, or the use of auto-labeling software such as Mechanical Turk. This has adverse effects on these DAL algorithms' performance, and most existing DAL literature focuses on noise-free settings. We explore existing literature around the problem of using DAL algorithms in the existence of label noise. We are particularly interested in the image classification domain using different deep representation learning frameworks such as convolutional neural networks (CNNs) and vision transformer networks.

In Section 2, we briefly discuss deep learning and the architectures used in image classification. Section 3, presents the main ideas behind active learning as well as the issues that arise when datasets contain noisy labels. In Section 4, we detail commonly used datasets for active learning on image classification tasks as well as the evaluation metrics. Section 5 is a detailed analysis of the literature on active learning with label noise in image classification tasks. We conclude by exploring possible directions for future research in DAL on vision tasks under label noise.

2 Deep Learning

Deep Learning (DL) refers to the use of artificial neural networks (ANNs) with multiple hidden layers [37], to approximate known or unknown functions. The multi-layered neural network was built on top of the perceptron [64] introduced in 1958. Over the years, different domain-specific DL architectures have been developed to enhance the quality of the learned representations from the different data modalities. Early research focused on improving optimization, custom layers and connections, activation functions, loss functions, and hyper-parameter tuning techniques for the multi-layer perceptron as a way to improve performance on different data modalities. For tabular data, tree-based ensemble learning algorithms such as Random forest [8], XGBoost [10], and CatBoost [61] are preferred over DL for their superior performance and resource efficiency. A non-exhaustive selection of interesting neural network adaptations to tabular data includes [66,63,60,4,6]. In the natural language processing domain, earlier work involved learning word and sentence representation using shallow neural networks in an unsupervised setting [57,55,26]. Until the wide adoption of attention-based transformer language models [77,67], word and sentence level embeddings are fed to a DL model with

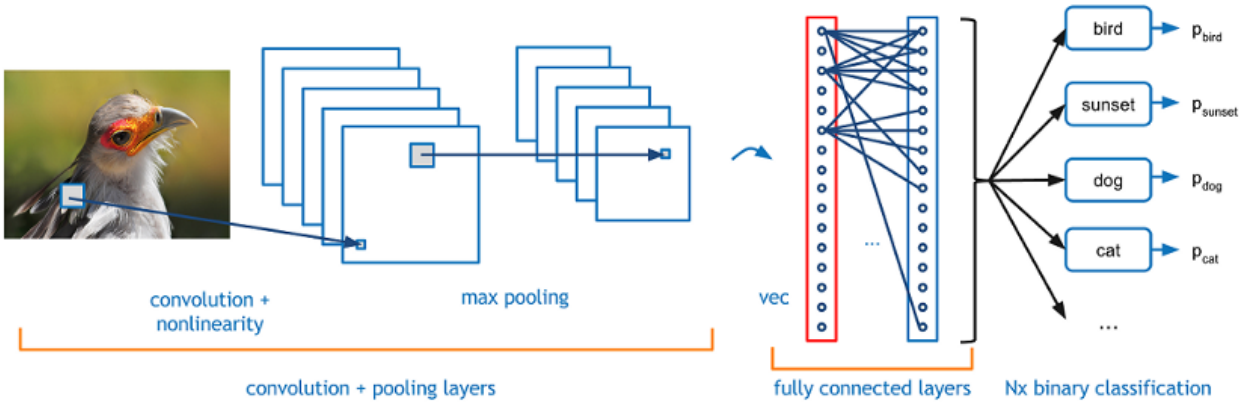

Fig. 1: [Source: Standard CNN]. CNN for classifying an image into one of the categories: bird, sunset, dog, cat, and more common objects.

recurrent connections such as a Long-Short-Term-Memory(LSTM) network [34] to achieve state-of-the-art results on down-stream text classification, sentence completion, named entity recognition or summarization tasks. For non-temporal visual tasks such as image classification, object detection, segmentation, and pose estimation [5], CNN-based architectures with specialized output layers and a lot of training data are still the most widely adopted approach. With each of these complex tasks, there are different challenges in the data annotation process that introduce varying levels of label noise.

While the DL methods discussed in this section have been applied to other supervised learning vision tasks such as detection and segmentation, we focus on approaches for image classification in this section. We give a brief overview of CNNs that are responsible for a large share of progress in vision-based tasks. We then highlight the use of more complex CNNs for image classification and finally explore the literature on state-of-the-art spatial attention-based models (Vision Transformers) in the context of image classification [39].

2.1 Convolutional Neural Networks

Convolutional neural networks were introduced by Yann Lecun and Yashua Bengio as an improvement to human-based feature extraction in training multi-layer neural networks on spatial data [42]. The key deficiencies with training fully connected feed-forward neural networks (FFNN) using back-propagation for computer vision tasks are efficiency and transformation (rotation, translation) invariance. Handling high-dimensional image data with standard input neurons is non-trivial and inefficient. Given that low-resolution image datasets are normally 28×28, the initial input layer using an FFNN would contain 784 neurons. If the subsequent hidden layer had as little as 100 neurons, the 2-layer fully connected network immediately has more than 78400 weights and bias terms connecting the two layers. The weights of the network are stored in high-dimensional matrices, and the flow of information in the forward and backward pass is performed using matrix operations. The number of input neurons and hidden layer depth required for accurate approximation of complex image-to-class mappings on high-resolution images using FFNNs is large.

CNNs are especially good at handling image data for three main reasons; firstly the convolution operation uses a sliding filter to identify and highlight the presence of local relations between pixels that represent important features. By so doing, we capture features expressing lines, edges, and corners implicitly. Secondly, in CNNs, the input image is not flattened into an array as is the case in using FFNN. This means the relative positions of pixels in a grid format are preserved and so we do not lose information through rearranging the pixels. Finally, in CNNs, filters have their own weights, but the same filter is used to slide over the image, and this means the features learned are invariant to the positions of patches on the image as the convolution operation learns local relationships between pixels. In addition, this way CNNs are able to use fewer weights than would be the case if we were to consider the absolute positions of pixels on the image and capture

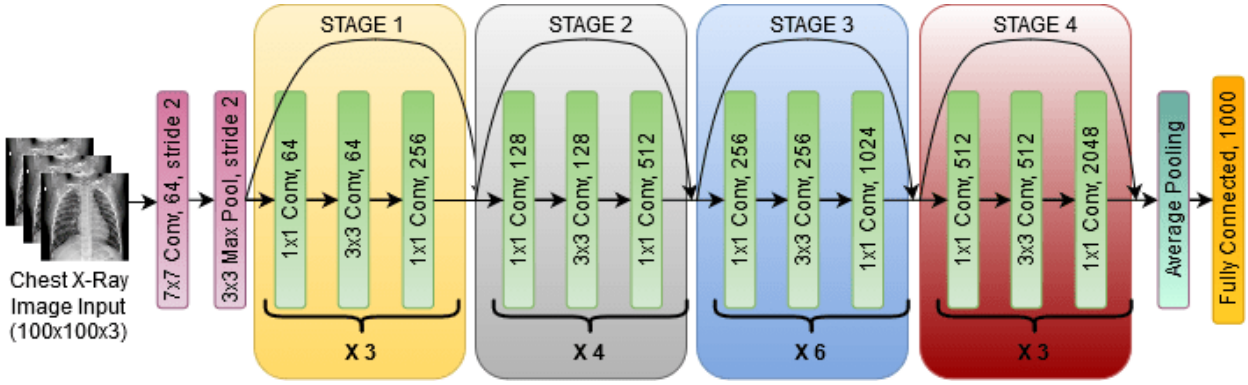

Fig. 2: Resnet50 architecture with convolutional blocks of different filter sizes and max pooling. Residual connections acting as memory cells arch above the blocks passing initial information all the way to the final layer [33].

positional encoding. Figure 1 depicts a simple CNN with convolution, pooling, and fully connected layers with non-linearity activation functions for image classification.

More advanced CNN architectures have been introduced, mostly similar in that they have multiple convolution and pooling blocks (earlier layers capture low-level features, and deeper layers capture higher-level features). The most notable of these are GoogLeNet [75], VGG [71], ResNet [33], DenseNet [35], and EfficientNet [48]. For example, ResNet, as shown in Figure 2, demonstrated the idea of residual connections (also called skip connections). The residual connections pass one layer's inputs directly to the next convolution block to provide lower-level context to the subsequent layer hence combating vanishing gradients in very deep networks. DenseNet on the other hand has a dense building block in that all the layers in a block have direct connections with each other, allowing for more effective reuse of features in the network. Also, by having all layers connected, a regularization effect is created so that the network does not learn redundant representations, hence combating over-fitting.

The different layers are connected by non-linear activation functions such as the popular ReLU and Elu [53,15]. CNNs have been the dominant approach to computer vision benchmarks for a large part of the last decade mainly due to their ability to extract meaningful spatial features from images. The main catalysts in ascending order of importance for this were the availability of large labeled training datasets, advances in computing hardware, and a reduction in the computational cost of training such Deep Neural Networks (DNNs). Post ImageNet [38], CNN-based models trained on very large labeled datasets have been used in the feature extraction and pre-training step of most fine-tuned state-of-the-art approaches in different vision tasks.

2.2 Vision Transformers

Before full transformer models in the language domain, the best LSTM models use a low dimensional vector representation to pass information from an encoder network to a decoder network, while using an attention mechanism. Attention in this setting is used to learn what parts of an input sequence are most important in predicting different parts of the output. In the original paper "Attention is all you need" [77], Vaswani et al. demonstrate that long temporal dependencies can be learned without the need for recurrence. The three fundamental components in a transformer network are a positional encoding of words, attention, and self-attention mechanisms. Positional encoding of both input and output tokens is achieved by assigning integer values to tokens/words based on their relative position in the input and output sequences. Unlike LSTMs, the work of learning word progression and relationships between input and output words is done implicitly by the network instead of designing networks with explicit bias in the form of recurrent cells and sequential processing. Self-attention makes it possible to learn good representations for most languages given a sufficiently large collection of text in a semi-supervised manner by masking tokens and letting the network learn what the missing word is in any given input sequence. The learned representations are then used on

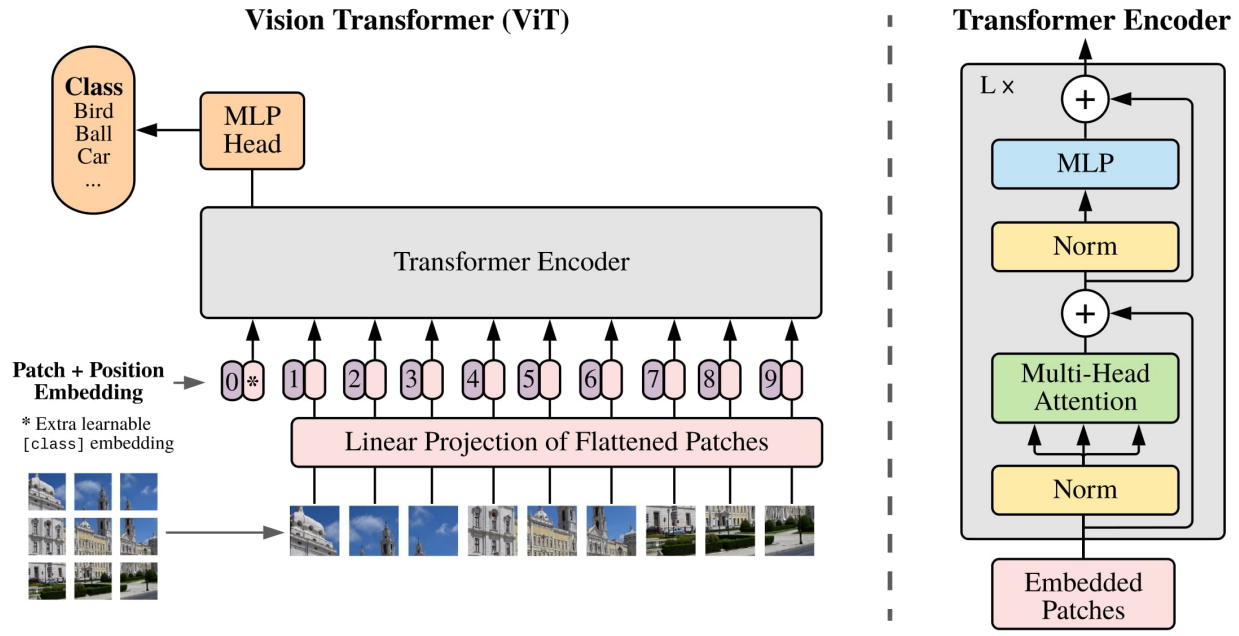

Fig. 3: Vision transformer architecture showing an input image split into 14 by 14 patches, and linearly projected to the standard transformer input space. The far right side of the image shows the components of the standard transformer block with multi-head attention [39].

a downstream task with fewer labeled data. Because transformers do not process input tokens in sequence, they are perfect for parallel GPU training.

Like most great innovations, the fundamental ideas of the transformer have been incorporated into CNNs [82,91,74], and in some cases completely replacing CNNs [76,47,12] to produce state-of-the-art results in various computer vision benchmarks. In [39], Kolesnikov et al. present the earliest vision transformer(ViT) to surpass state-of-the-art CNNs on most image classification benchmarks. They show that in the large dataset regime, ViTs achieve higher classification accuracy, are more computationally efficient, and show no signs of saturation compared to CNNs such as ResNet and EfficientNet on increasingly larger datasets. The main difference between the natural language processing (NLP) transformers and the vision transformers is in how the input is encoded. With vision transformers, they take 14 by 14 patches from an image, flatten them, and apply a linear projection onto a higher dimensional space equal to that of the original input space of the NLP transformer. The spatial proximity relations of patches are implicitly left to the transformer to learn in the following way: They add a trainable 1D positional encoding vector to each patch's linear projection. The positional representations are organized in the order of the patches starting from the top left corner to the bottom right corner of the image as depicted in Figure 3. Beyond input encoding, the rest of the ViT architecture is similar to that of the language transformer for classification tasks.

The paper shows interestingly that, through the attention mechanism the transformer layers are able to learn the same low-to-high level features with increasing depth as is the case with deep CNNs. Other notable implementations of ViTs for image classification without label noise include [90,12,47]. ViTs are included in this review and further discussed in Section 6 as we perceive them to be a very important area for future research. This is because they are progressively becoming the dominant multi-modal approach and yet very little work has been done in applying them to DAL and learning with label noise for image classification. These models are designed in a modular fashion to easily be able to learn both language and image representations for image captioning, classification, scene-text understanding, and visual question answering [90]. It is particularly interesting since the authors present a joint contrastive loss (image-to-text and text-to-image), image classification loss, and image-to-language captioning loss, allowing for efficient training of a single

network for multiple tasks, and the ability to transfer the learned representations to a different downstream task and dataset.

In the next section the active learning (AL) framework for machine learning is described, including key approaches for training deep learning models on a labeling budget in the case of clean labels, and finally, the scene is set for label noise and the literature addressing DL on noisy labels.

3 Active learning

In most supervised machine learning use cases, there is an initial data collection and labeling cost, in both money and time. In some domains and tasks, datasets are inherently difficult to label for a variety of reasons, meaning more time is needed even by an expert human annotator to assign a label to each sample. In other cases the cost of hiring expert annotators is high, such as is the case in medical imaging [25,40], or the cost of producing the samples is high, such as is the case in experimental physics where observations come from very expensive telescopes or particle accelerators. This presents a challenge to the real-world use of machine learning systems, especially as unlabeled dataset sizes increase. Active learning is a machine learning paradigm, as depicted in Figure 4, that seeks to address this problem by letting learning algorithms iteratively select a subset L^m of size m, from a larger unlabelled dataset U^n of size $n : m \leq n$, to be labeled by an oracle O for training. The active learning mantra can be stated as follows: Train a machine learning model on a significantly smaller labeled dataset, with little to no drop in test performance, all the while staying within a pre-determined labeling budget B.

Deep active learning algorithms (DAL), while overlapping, can broadly be grouped into pool-based methods, density-based methods, and data expansion methods. Pool-based methods select samples for labeling from an unlabeled pool, based on either the uncertainty of the currently trained model on samples U^n, the diversity of samples in the labeled set L^m used to train the current model or a combination of both [44,50,17]. Pool-based methods are simple in their formation and implementation but can be computationally expensive for large datasets of high dimensional data such as images. Since pool-based methods largely rely on metrics evaluated on the entire unlabeled dataset to select new candidates, this is not ideal for applications that require low latency. Density-based methods seek to capture key characteristics of the underlying data distribution. This is done by selecting a core-set of samples for labeling that are sufficiently representative of the entire dataset, and leads to good generalization [68,59,58]. More recent literature blends pool and density-based methods to take advantage of each approach's benefits. These methods thus lead to efficient and robust models trained on core-sets containing diverse samples that maximize the margins between object classes [30,22]. Some methods in this approach use the hidden layer representations from training a self-supervised task on the image data, instead of the raw pixels. These include pre-training on image orientation, random $(90, 180, 270, 360)^\circ$ rotation classification, or self-supervised contrastive learning, where the target is an arbitrary patch of adjacent pixels in the image [11,18,78]. Data expansion methods seek to expand the training dataset, by generating reasonably realistic synthetic data samples for each target class to enhance the learning algorithm's performance on the real test dataset [11]. Since their introduction, Generative Adversarial Networks (GANs) and their variations [24,23,73] were the go-to method for generating synthetic data. However, the training of GANs is unstable, the samples tend to be unrealistic, and it is hard to evaluate these samples for quality [7,51].

3.1 Label Noise

label noise refers to the scenario in which data labels are corrupted, with or without intention, so that we do not have 100% confidence in their correctness. Label noise is different from feature noise which is normally used to refer to adding gaussian noise to feature values. Label noise impacts learning algorithms more adversely than feature noise does, and is harder to deal with [13,80,1,16]. Label noise is inherent in the data collection and processing life-cycle. Most real-world datasets are subjected to a number of label noise sources based on how the data is collected, curated, and stored. Label noise in practice broadly stems from (1) incorrect crowd-sourced labels where the annotators are non-experts such as is the case with [81], and [2], (2) incorrect expert annotations due to the complexity of the data, as is common in medical fields [25], (3)

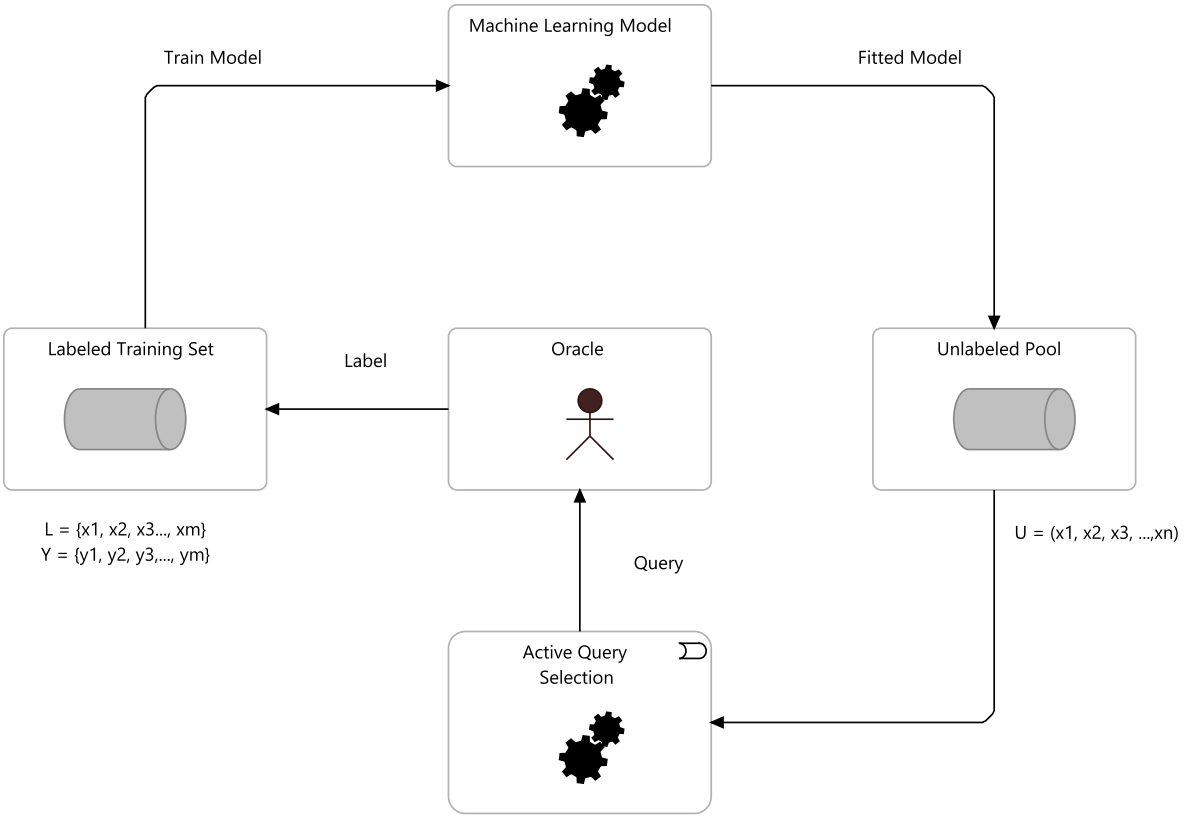

Fig. 4: The five main components to the standard Active Learning Framework. Each of these components may vary depending on the complexity of the data to be learned and the available resources. Most work in active learning has focused on the development of query selection algorithms that lead to highly informative and diverse data samples for labeling by the oracle.

labeling errors introduced by automatic labeling by web crawling software and other AI labeling systems such as [65], (4) noise introduced by multiple experts or non-experts labeling the same sample differently.

Learning noisy labels is especially hard due to the fact that cost functions are generally significantly less complex than feature extraction layers. Label noise can be grouped, and is mostly treated based on what is known about the noise-generating distribution [52]. Some datasets contain label noise from a known and quantifiable generative distribution, while in other cases, too little or nothing is known about the noise transition matrix to model. Label noise can be class-independent or class-dependent. Class-independent label noise is the easiest to generate. The generative process can be summarized in this manner: for each sample, the class label is replaced with a random class label, with a fixed probability $1/N$ where N is the number of classes [56]. Class-dependent label noise is normally a result of expert human annotation. It results from pairs of very closely related or indistinguishable classes being occasionally mislabeled [28]. For example, the true large-sized cat is occasionally labeled as a small dog, and visa versa. Common methods for training DL models include first filtering out samples with a high probability of being noisy and iteratively training on a dataset with trusted labels until a threshold is reached. The filtering process in most literature involves training two different neural networks with a custom loss, and monitoring samples on which they disagree on predictions. This method works well since it has been shown that the networks train on stronger signals first, which is the case in a dataset with predominantly clean labels. Representative methods in this approach, trained in a non-active learning manner include Decoupling [49] and Co-teaching [29]. The main implementation difference between the two approaches is in how the two networks' weights are updated. Decoupling updates each network's weights based on its prediction error when the networks have a prediction disagreement. Co-teaching on the other hand, cross-updates the weights with the error signal from the other network. Unlike Decoupling, Co-teaching addresses noisy labels explicitly by enabling the networks to peek into each other's hidden state, simultaneously reducing the risk of each network over-fitting the noisy input.

We have introduced deep learning, active learning, and learning with label noise. For further reading, we suggest the survey papers [1,62] on image classification with noisy labels, and DAL on clean labels respectively. DAL methods on noisy labels are presented in Section 5.

4 Evaluation Datasets and Metrics

In this section, we introduce datasets and evaluation metrics commonly used for active learning and learning with label noise. The State-of-the-art DAL methods on zero-label noise datasets are [25,40,68,59,58]. Datasets and their meta-data are provided that pertain to AL and DL on label noise. We conclude the section by exploring the evaluation metrics for DAL for noisy data on common bench-marking datasets.

4.1 Datasets

As stated previously, the meteoric rise of DL algorithms was largely due to the availability of large labeled training datasets. In image classification, the most notable datasets include MNIST [43], ImageNet [38], CIFAR-100 [41], CALTECH-101 [20], SVHN [54], and MS COCO [46]. Public evaluation datasets facilitate a centralized evaluation of algorithms on a pre-defined task. These datasets can be downloaded from their websites or the different DL frameworks such as PyTorch, TensorFlow, Jax, and Theano. The best-performing models and their results on these datasets are normally hosted on a public leaderboard for the dataset. When evaluating datasets for DAL on image classification tasks, the standard practice is to use the same datasets as in full dataset image classification, but we monitor performance gain after a pre-defined number of labeled examples. While in practice this may not be the case that all labels are available upfront as is the case in using fully labeled datasets, the training cycle of DAL algorithms applied on these complete datasets substantially mimics the process of obtaining labels from an oracle for a live stream of unlabeled data within a budget.

The datasets vary widely in size, the number of classes, and the complexity inherent in telling the classes apart. Of all the commonly used AL datasets, MNIST is the least complex, with only 10 classes of hand-written digits in single channel 28×28 images. CALTECH-101, ImageNet, and CIFAR-100 are higher-resolution image datasets. These datasets contain more classes than MNIST, some of which are harder to

tell apart. In passive learning, the model sees all available training samples per class, but in the DAL setting, depending on the query algorithm and scarcity of a class, the algorithms may never see more than a third of the samples of certain under-sampled classes. This leads to poor validation performance.

Large-sized datasets with high-resolution images also pose a computational problem in DAL algorithms that select diverse samples based on a distance measure to all other unlabeled images. This can be extremely costly to compute in both time and hardware resource requirements. For these reasons, the reported performance of DAL algorithms on these datasets is lower than that of non-active learning algorithms since researchers have a low incentive to test complex DAL algorithms on large datasets. Table 1 contains a non-exhaustive list of commonly used image classification datasets for active learning and learning with label noise.

Dataset	Year	# Samples	Classes	DAL Papers	label noise Papers
ImageNet	2012	1,431,167	1000	[86]	[31]
SVHN	2011	660,000	10	[27,79,68]	[27,83]
MNIST	2010	70,000	10	[45,32,27]	[88,27]
CIFAR(10,100)	2009	60,000	(10,100)	[18,88,80,45,70,32,27,89]	[88,28,31,27,89]
Caltech101	2004	9,000	101	[45,86]	-

Table 1: Image classification datasets commonly used for deep active learning and the training of DL with noisy labels

ImageNet and SVHN, being the larger of these datasets, are not well suited for DAL because training a single DL model on a large dataset is computationally expensive, and takes a lot of time. The computation complexity is worse in the case of DAL algorithms due to the iterative nature of the process. Retraining a large model over and over on the ImageNet or SVHN datasets is time-consuming. This is reflected in the literature by the reluctance of authors to use these large datasets for DAL classification, in favor of relatively small datasets such as CALTECH-101 and CIFAR-100.

Developing and training algorithms for handling noisy labels follows one of two paths: using datasets with noisy labels introduced by one or more of the noise sources listed in Section 3.1, or noise-free datasets to which measurable label noise is injected by perturbing existing trusted labels. In the existing literature, the same datasets (MNIST, CIFAR-100, CALTECH-100) commonly used for image classification are used for noisy label classification, with a pre-determined probability of swapping each label. This probability is also called the noise rate, and the higher it is, the more corrupted the dataset becomes after noise injection. In literature, it is common to inject $30\% - 60\%$ random symmetric label noise before training, while keeping a test set that is free of label noise.

Datasets such as ImageNet with a large number of classes (1000) tend to also not be favored for the purpose of evaluating DAL methods that address learning under label noise. The reason here is that, with more class labels, the likelihood of class-dependent labeling errors at the time the dataset was created is higher. The kind of deliberate noise injected into datasets for noisy label learning is class-independent and is measurable as opposed to class-dependent noise that may be inherent in the data collection and annotation process. Class-dependent label noise makes training DL models harder, and there is lower confidence in the correctness of the test set labels used for evaluation.

4.2 Evaluation Metrics

The general evaluation methodologies for DAL algorithms on noisy labels are the same as those used for standard datasets for image classification. Top-1 accuracy is the most commonly used metric. Not much consideration is given to the underlying class distribution in most existing work and so it would be of interest to explore how class imbalances affect DAL algorithms in the presence of label noise. Since DAL is also concerned about performance under a budget, it would make sense to measure budget efficiency, a measure commonly not well documented in the literature.

Given that active learning involves training a model a couple of times for every batch of labels received, it becomes obvious that the computational cost of DAL algorithms should be a big consideration. In [87], Yoo et al. propose the use of a small network for performing query selection so that the retraining and labeling cycles run faster. Once the labeling budget is exhausted, a larger and more powerful network is then trained using the obtained labels. While this approach can be very useful in scenarios where time is of the essence, it has a big drawback. The weakness is that using a weaker learner for sample selection could lead to lower sample diversity since a weaker learner does not perform a very good job of understanding the boundaries between classes in the feature space.

In [72], Signha et al. demonstrate the use of transfer learning for fast extraction of useful representations in a DAL setting. They show that using large pre-trained models and only fine-tuning the DAL task achieves good results with considerably fewer labeled examples. This means that, given the same budget, their approach has a higher label efficiency than a model trained from scratch. This also means for a given target performance, they require less computational resources and time to fit the target, by leveraging good pre-trained model weights. The work of Settles [69] contains a comprehensive survey of the computational cost of active learning algorithms. The main findings in this work are that the cost is influenced largely by the dataset size in terms of both the number of samples and the size of each sample. Settles also states the complexity of the query selection algorithm as well as the number of samples per batch are big factors in the total computational cost of DAL. In the case of DAL with label noise, it is critical that we have a good understanding of the underlying label noise so that the test set remains clean. While this works in developing DAL algorithms on well-known datasets, it remains unclear how the test set integrity is guaranteed in practice. If the correctness of the test labels cannot be guaranteed, evaluation methods such as top-1 accuracy, precision, and recall do not offer any reliable measure for the network's generalization performance.

In this section, we explored datasets and evaluation metrics commonly used in comparing DAL algorithms, in particular under the setting of label noise. The next section is the main focus of this work. We explore methods leveraging the versatility of deep neural networks in the active learning framework where labeling budget is an important metric, and we are faced with a noisy label challenge.

5 Deep Active Learning Algorithms for Noisy Labels

In this section, we focus on the main contribution of this manuscript: exploring literature on DAL algorithms used for image classification in the presence of label noise. It is worth stating that while literature is rich in theoretical approaches for handling label noise in the offline setting, very little has been done for active learning algorithms. Methods that address label noise by modeling the underlying generative distribution and filtering noisy label examples from the training set to achieve better performance are few and in between. We foresee a lot of work going into this work and look forward to understanding how iterative processes best approximate a noisy label distribution. We are also interested in understanding how low sample numbers affect label noise distributions. These ideas remain unexplored in literature.

The methods in this section are predominantly independent of the noise distribution and seek noise-robust active training by using customized model architectures, loss functions, or training procedures. In [27], Gupta et al. propose the use of standard sample diversity and importance query policies, supplemented by the model's confidence scores on samples. They argue that DNNs are normally uncertain about the decision boundaries between classes very early in training. Training with label noise exacerbates this problem since temporary and imaginary boundaries could form based on mislabeled samples, and through diversity sampling, get propagated into important query batches that influence model uncertainty and sample importance. The authors use the BALD score, introduced in [21] (not to be confused with the paper [9]) as an importance score to ensure the information content of samples for labeling per batch is optimal and a good representation of the entire dataset. The inclusion of model uncertainty in the sample selection query ensures labeled batches will include samples the current model is very uncertain about, and these, assuming satisfactory oracle label accuracy, improve the entire DAL cycle under label noise.

The inclusion of highly uncertain samples is only one-half of the novelty of their approach to robustify learning under a noisy oracle. They include a denoising layer to their network. The denoising layer is explained in the following manner: The softmax output of the original classifier is fed to the denoising layer, and the model is

trained on the denoising layer, which represents a non-zero probability of predicting a particular label given the true label. This final denoising layer's weights, unlike normal final softmax outputs, are constrained to ensure noisy labels have little impact during training. During testing, the penultimate layer's output is used instead of the denoising layer for prediction. In this way, the denoising layer serves as a rigorous teacher to the student, becoming a noise-robust model used in testing and deployment.

Gupta et al. show that their method, in the noise-free setting achieves similar performance to the common baseline DAL methods, such as the original BALD [9], core-set, entropy-based selection, and random sample selection on the MNIST, CIFAR10, and SVHN datasets. We attribute these results purely to the addition of model uncertainty to diversity and information gain in selecting samples. We argue the denoising layer as described in the paper would serve no purpose if the oracle provides only clean labels and so the results in the noise-free setting would be better stated as: "no performance loss or gain" from adding the denoising layer in the noise-free setting. At 10% and 30% label noise, their method outperforms the reported standard benchmarks on all of the three datasets, speaking to the effectiveness of their denoising mechanism. While these are good results, they fail to demonstrate how the approach compares to similar state-of-the-art DAL methods tailored for noisy labels, and how adding the same denoising layer to DAL ResNets trained under entropy only or random sample selection compares to their approach. The paper also lacks details on the training setup that is important for reproducibility, such as the hardware used, the exact deep learning architecture, whether pre-trained weights are used or not, and the number of training epochs.

Similar to [27], in [88], Younesian et al. introduce a DAL framework (DuoLab) for training on noisy labels using weak and strong oracles. CIFAR10 and CIFAR100 are used in training and testing a CNN, with 30% and 60% label noise. They adopt similar criteria to [27] for query selection, namely using information gain and uncertainty. The weak and strong oracles refer to the innate differences in human labelers' generalization abilities and labeling quality. It is assumed that weak oracles are cheaper and more likely to produce incorrect labels than strong oracles, and so this approach's novelty is particularly more interesting in the real-world setting where there is always a need to reduce labeling fees paid to the oracles. However, the use of two oracles seems to not affect the overall performance of the DAL classifier in their work, but rather gives budget subsidies.

When it comes to dealing with noisy labels, instead of robustifying their model, Younesian et al. approach the problem by filtering out samples suspected to have noisy labels. Their DAL approach starts with an initially labeled dataset used to train the classifier, but they deviate from the conventional use of a random batch of samples to perform this initial training. Their overall approach hinges on a key and possibly flawed assumption that there exists a small and clean batch of training examples that can be used to initially train the model. While in practice we can optimistically assume it is possible to push physical data and labeling boundaries so this initial clean set is available, the paper lacks the minimum theoretical guarantees in the case we are unable to say with "100%" certainty that specific labels from the oracle are correct. If we were to initially assume some labels are "100%" correct, the paper does not make it clear as to how the correctness of such labels is verified given that the oracles have a noise rate of up to 60%. Once the model is trained on the initial clean samples, the model predicts the classes of all samples in the unlabeled pool, and the model's confidence score on the top 2 classes is used in deciding whether a sample is noisy or not. They measure the difference in top-2 class probabilities for each sample, and declare the top-k samples with the lowest margin as potentially noisy and so not fit for training the network.

It becomes obvious that this approach will lead to a number of false positives and false negatives, and are likely to affect the overall performance of the model in classes that are hard to tell apart. Another issue is that a model trained on a very small subset of a large dataset can display a high top-1 class prediction confidence while being totally wrong if it has seen very little to no examples of a class in training. They combat this by reusing the noisy labels once all the clean examples are exhausted. The noisy samples are clustered based on the trained model's penultimate representation of each noisy sample. The samples within each cluster are then ranked based on their informativeness, and the top K samples per cluster are picked and used to further train the model. The training batches on these samples are not random. They are ordered so that the most informative samples are in the first training batch and the least informative are in the last batch. It is unclear how this process circumvents the need for actual ground truth labels, and how this further training on noisy labels does not negatively affect the performance of a model first trained on clean labels. Reported

test accuracy results show that DuoLab outperforms noise-resilient baselines on both CIFAR 10 and 100 with 30% label noise.

In [89], Younesian et al. propose QActor, an approach that follows the same idea as their earlier work in [88]: Identifying and reusing possibly noisy labels. Over and above this, the paper introduces a noise-aware informative measure, and they demonstrate for the first time how dynamic allocation of the labeling budget per query leads to better performance as compared to the convention of equal distribution of the budget across the number of query cycles. In this paper, it is assumed that a small set of clean initial training data, as well as a clean test data, are readily available. The DAL algorithm decides which samples are to be sent to the oracle for labeling. They also leave it to the DAL algorithms to decide how many of the samples fit the selection criteria and are labeled per iteration. On each iteration, a batch of highly informative samples as measured by entropy is sent to the oracle for labeling, and then the labels are compared to the current model's predictions. Samples, where there is a disagreement, are sent to a suspicious data collection. These samples are later ranked on informativeness, and resent to the oracle for relabeling, with the previously assigned label kept in mind. The authors argue that relabeling samples that the model is uncertain about and are potentially incorrectly labeled by the oracle leads to the highest gain in model performance since these are likely samples from classes that are easy to mix up.

In [36], Huang et al. approach DAL under oracle noise in a way that is different from most of the work discussed thus far. The complexity analysis performed in their work proves DAL is possible and viable with oracle epiphany. They explain that empirical evidence has shown that oracles are likely to delay providing labels on samples they are unsure about until more related examples are presented to the oracle, at which point they have an epiphany, and are able to provide a more confident label to such examples. Having had the experience of hand-labeling 1000 images from a large fisheries dataset, we agree with the authors that oracle abstention and epiphany are realistic considerations. An interesting idea mentioned in the paper is adding one more possible class label for a classification problem with N classes so that the oracle has $N+1$ possible classes to choose from. This class label can be one of: "I don't know" or "unsure". Adding this choice relieves the oracle of the urge to guess between two or three most likely label assignments for a hard sample image, hence increasing the overall confidence in the correctness of the class labels that are actually provided. This is true since in a perfect world the oracle has to be the source of absolute truth, and so we are not interested in cases where the oracle guesses correctly. An oracle guessing would be useful if they are allowed to provide a measure of how confident they are of their guess.

In their approach, they use Markov chains to model when epiphany occurs for a certain class, that was previously hard to label to the oracle. The two main assumptions made in this work are: 1) The oracle is honest, and 100% accurate on the samples he/she decides to label anything other than "unsure". 2) Given the oracle is honest, all samples avoided will have correct labels once epiphany occurs. For both assumptions, the authors further assume once an epiphany occurs, no drastic changes in the oracle's assignment of labels will occur. While these assumptions sound reasonable, it is worth stating that in the real world, a time-constrained oracle optimizing for earnings is unlikely to avoid assigning labels to samples they are unsure about. This is especially true if they learn earlier on that samples they label "unsure" will always return, requiring more of their time. The assumption that the oracle is 100% accurate on examples they label is also very unrealistic and does not factor in the wide-ranging spectrum of human capabilities in labeling. It would be more useful if the authors stated how 100% oracle label accuracy would be guaranteed or tested post-epiphany. Comparing this approach to the works [89,88] of Younesian et al. a spectrum of computer-human involvement in label noise filtration can be drawn. At one end Huang et al. use the oracle as a sole decider of label uncertainty, and in a more hybrid setting, Younesian et al. use the currently trained classifier's confidence score together with the oracle's label as a filter for potentially noisy labels. The model-only approach to noise filtration is using the confidence score margin of the top-2 predicted classes as used in [27]. In both Younesian et al. and Huang et al.'s approaches, the "uncertain" samples are sent back for relabeling.

In [3], Amin et al. present the dual-purpose learning framework. While they do not explicitly focus on nor address label noise, their combined DAL, and abstention learning approaches provide valuable insight into the intricacies of DAL and abstention, which are important components that are not discussed as rigorously in [36]. The two bodies of work also differ in that Amin et al. investigate the generalization bounds of the DAL and abstention setting and find interestingly that even with an unlimited labeling budget and no labeling

noise, the upper bound on the observable generalization loss that exists in the passive learning case can not be guaranteed while oracle abstention is at play. The authors however do not provide empirical comparisons of their method to state-of-the-art DAL algorithms, nor do they detail how their unique dual method would impact performance when applied to such highly-performant algorithms. The work of Yan et al. [85] is a more general approach to the work of Amin et al. since they consider DAL under imperfect labelers, and allow for abstention. Yan et al. go on to show that under strict assumptions on the dimensionality of the decision boundary, abstention, and noise rates close to the decision boundaries, their method generalizes the lower bounds on algorithms such as [3]. A significant contribution of their work is in demonstrating that their algorithm need not be aware of either the label noise rate or abstention rate. With restrictions on how label noise and the rate of abstention are distributed around the decision boundaries, their algorithm performs significantly better than prior methods.

6 Conclusion and Future Research Directions

To summarize the literature: (1) A lot of work has been done on training DL models on noisy labels in an offline setting. (2) The literature on DAL methods is also very rich in the case of no label noise. (3) While this is an area of research that has very practical applications and financial impact, there is very little work done at the intersection of DAL and label noise. It is worth noting that the power of ViTs has not been explored in DAL as much as it has been in fully labeled image classification datasets, and so we see huge potential in improving DAL by leveraging unique characteristics of ViTs in image classification tasks. The exploration of the transformer layers and how attention can be used in understanding diversity, importance, and uncertainty is especially intriguing to us. In Section 3, works using self-supervised pre-training are presented that attain good lower-dimensional representations of the images. These have been shown to lead to a good core set of samples for labeling in an active learning framework. It is of high importance that contrastive learning methods are explored further as methods for deriving good image representations that can then be used in improving the diversity, importance, and uncertainty-based selection in queries sent to an oracle for labeling.

In terms of DAL for noisy labels, the works of Younesian et al, Huang et al. and Gupta et al. described in Section 5 represent methods and ideas with substantial impact in DAL. In all these approaches, filtration of noise is well addressed, but with a lot of questionable assumptions. Establishing unified DAL and label noise benchmarks and datasets would add a lot of value and ensure future methods conform to realistic assumptions about the training process and the oracle. In all methods addressing this problem, only CNNs are used and the performance is only compared to baseline DAL methods as opposed to a more convincing comparison to state-of-the-art DAL methods. We would like to be able to perform the same analysis on DAL methods for label noise using ViTs against the best DAL methods. This is especially important since ViTs have been the most dominant architectural choice for image classification in recent times. It is also critical that substantial effort is put into understanding how key assumptions made in DAL methods for noisy labels affect results, and establish convergence and complexity guarantees mathematically. This means that it is not adequate to only understand DAL on noisy labels through experimentation, but also some effort needs to be put into establishing convergence guarantees at a theoretical level.

In all existing domain literature, to the best of our knowledge, the oracle gets no feedback. It would be very interesting to explore how a cyclical feedback loop between the oracle and the model improves both of them. Intuitively we hypothesize it would help to have tips generated by the model in an unsupervised manner accessible to the oracle in cases of difficult samples. A step closer to the ideal interaction is allowing for abstention and epiphany. One way to do this is through contrastive learning. We hypothesize that, since contrastive learning models, through rotation or patch prediction for example, can be trained with no actual problem domain labels and thus noise-free, the representations learned from this step can be used in proposing similar samples to an oracle before they can abstain from labeling a sample. In this manner, the oracle may reach epiphany much earlier than they would in the frameworks listed in the literature. We would also consider answering the question of whether a real human annotator with a known or unknown noise rate can improve, and by how much in this setting.

Since reproducibility and robustness are key factors in the ongoing development of DL and AL, we are interested in performing extensive training of out-of-the-box DL models in the active learning framework in the presence of label noise. This will not only ensure researchers know what to expect out of different models without any hyperparameter tuning but also set up benchmarks that are specific to DAL with label noise. We are interested in covering as many image classification datasets as possible, using multiple CNN and ViT-based models, different AL algorithms, different noise handling algorithms, as well as different loss functions. This is vital to the field as it encourages clear and concise statements about the mode and training conditions, and ensures future methods are to be compared on a level playing field.

Lastly, a key consideration is in computation. Most work in the area does not explicitly state and discuss the computational complexity of the methods. In a world ever so gravitating towards lowering carbon footprint, it is important we are able to assess DAL methods on noisy labels not only on their accuracy but also on their computational requirement. This is an interesting avenue of research if one considers how recent work on training large language models has shown there are considerable trade-offs and gains to be made in the computational operations required, model performance, architectural choices as well as representation size. Some of these have gone in the face of established results [84,14].

7 Literature Networks

Below we present a visual depiction of the literature most related to this review focusing on deep active learning on noisy labels for image classification. We present the views as images of network graphs where the nodes represent papers and the edges represent the similarity between articles. The larger the node, the more influential the article is to related articles, and the thicker the connection between any two papers, the more closely related the articles are. The Connected Papers visualization tool [19] was used to create these graphs.

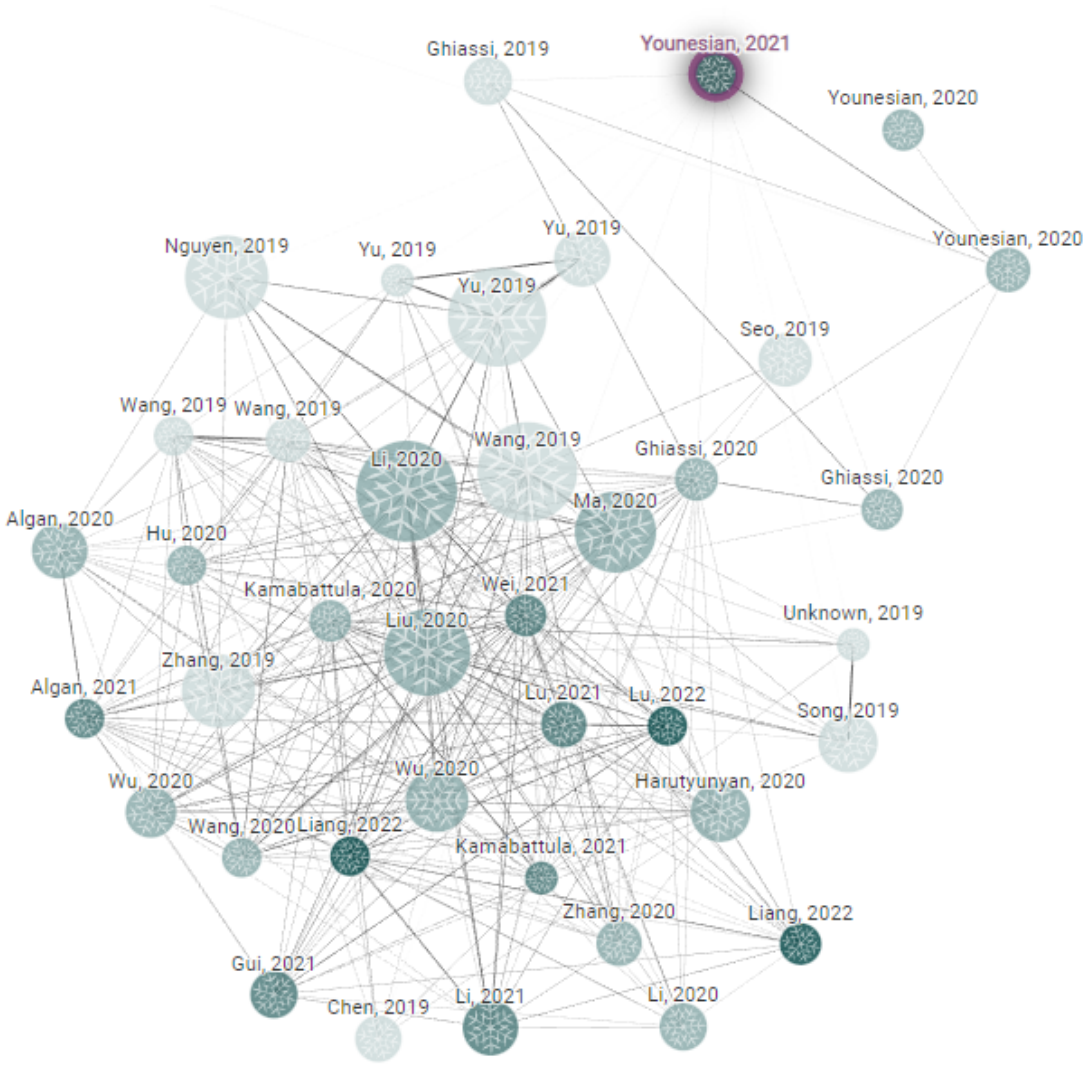

Fig. 5: Deep active learning with noisy labels literature closely related to Younesian et al. [89]

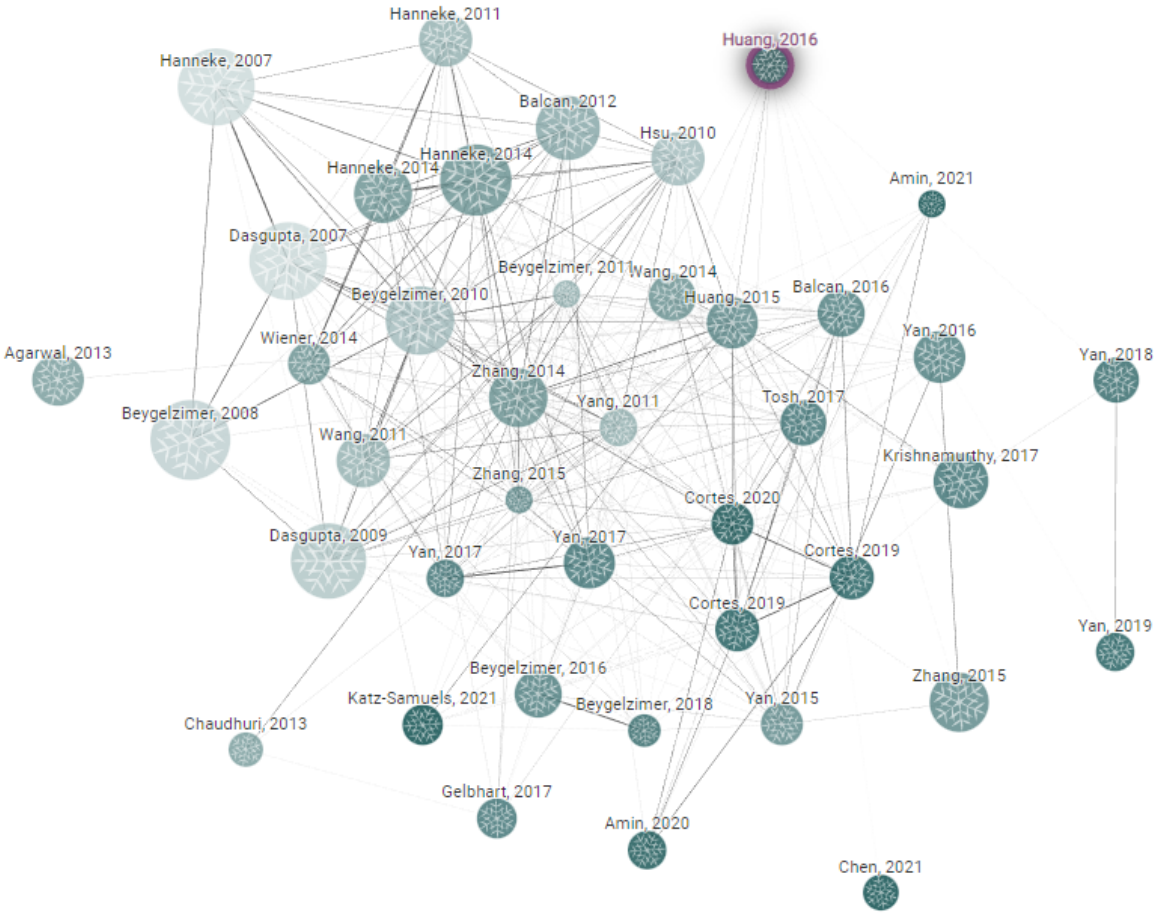

Fig. 6: Deep active learning with noisy labels papers related to the work of Huang et al. [36]

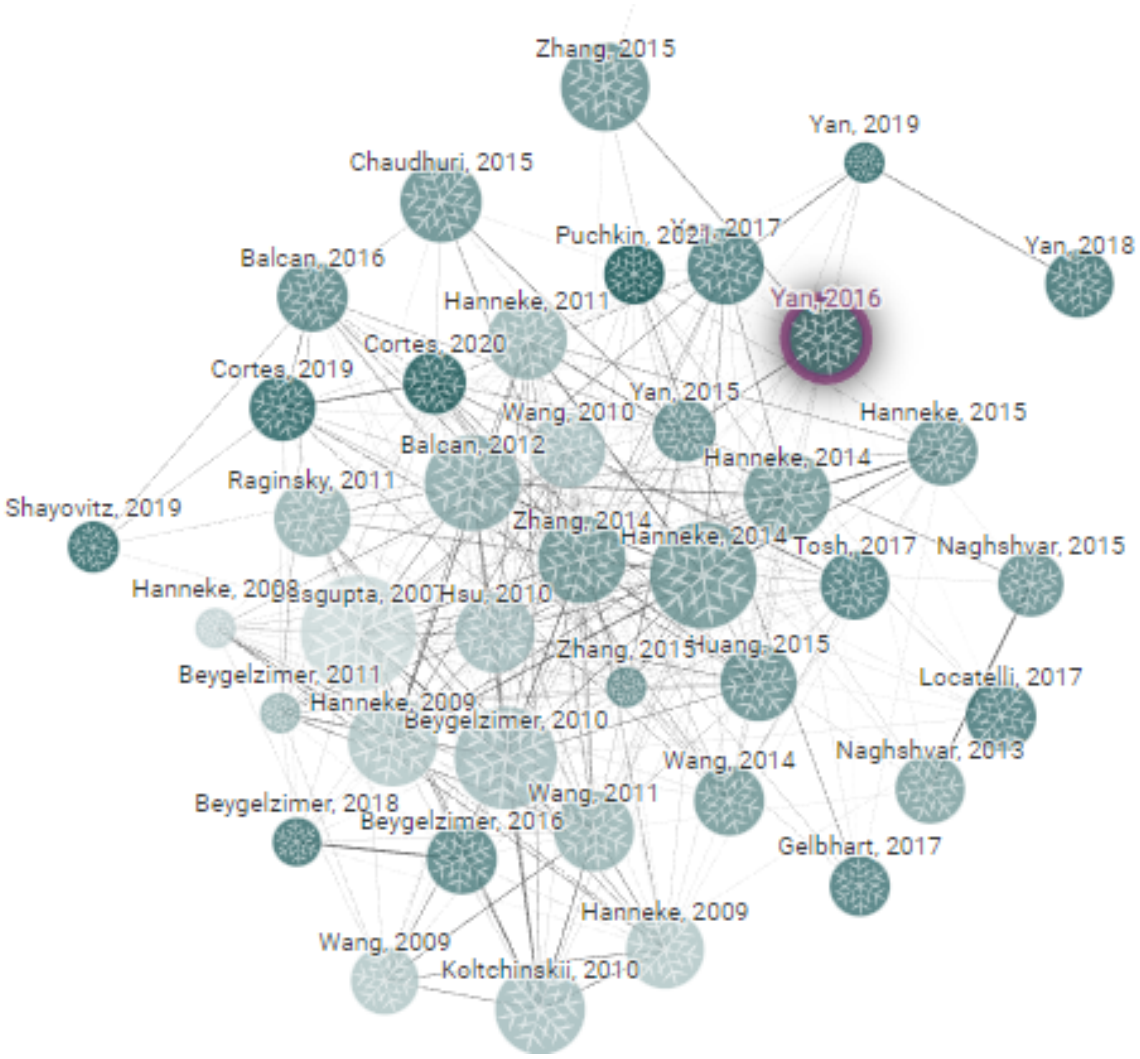

Fig. 7: Deep active learning with noisy labels papers related to the work of Yan et al. [85]

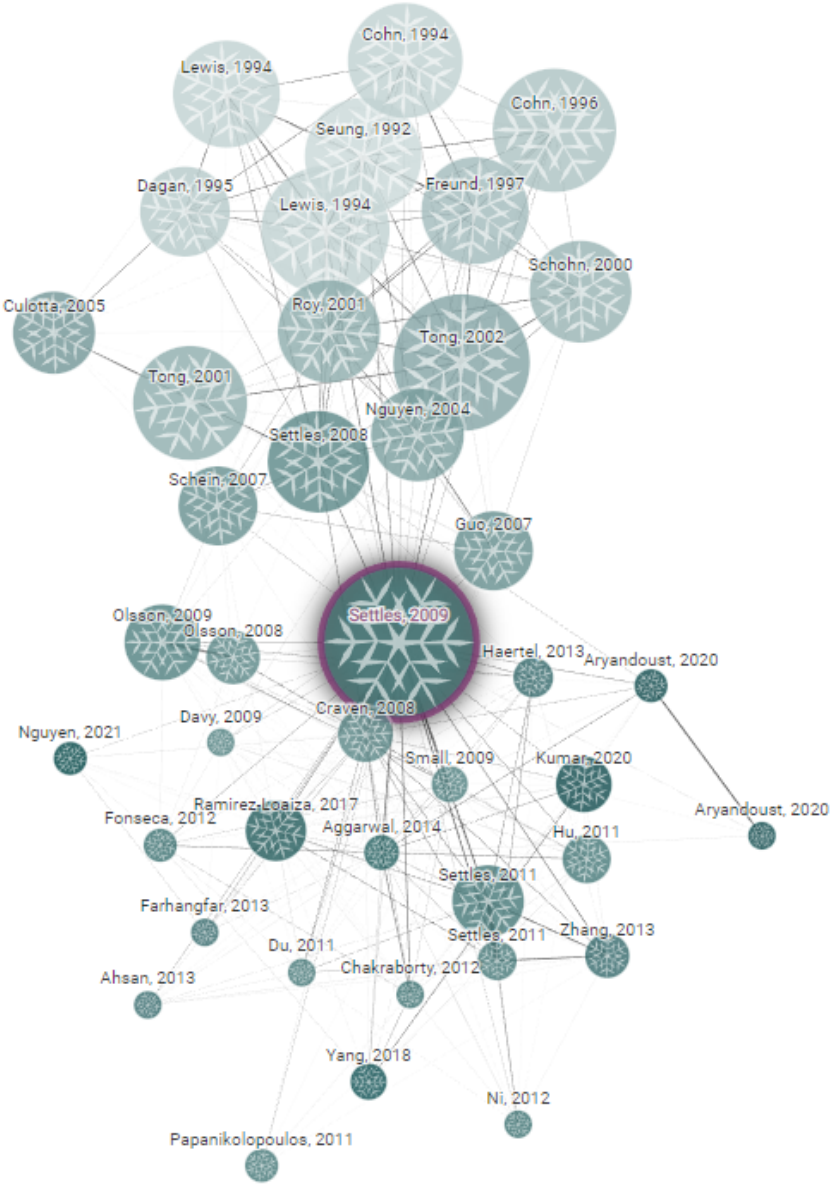

Fig. 8: Active learning work closely related to the survey manuscript by Settles [69]

References

1. Algan, G., Ulusoy, I.: Image classification with deep learning in the presence of noisy labels: A survey. ArXiv **abs/1912.05170** (2021)
2. Amazon: Amazon mechanical turk (2022), https://www.mturk.com/
3. Amin, K., DeSalvo, G., Rostamizadeh, A.: Learning with labeling induced abstentions. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems. vol. 34, pp. 12576–12586. Curran Associates, Inc. (2021), https://proceedings.neurips.cc/paper/2021/file/689041c2baed0f6d91050495d632d6e0-Paper.pdf
4. Arik, S., Pfister, T.: Tabnet: Attentive interpretable tabular learning. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 6679–6687 (2021). https://doi.org/10.1609/aaai.v35i8.16826
5. Artacho, B., Savakis, A.: Unipose: Unified human pose estimation in single images and videos. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7033–7042 (2020)
6. Baohua, S., Lin, Y., Wenhan, Z., Michael, L., Patrick, D., Charles, Y., Jason, D.: Supertml: Two-dimensional word embedding for the precognition on structured tabular data. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 2973–2981 (06 2019). https://doi.org/10.1109/CVPRW.2019.00360
7. Barnett, S.: Convergence problems with generative adversarial networks (gans). ArXiv **abs/1806.11382** (2018)
8. Breiman, L.: Random forests. Machine Learning Journal **45**, 5–32 (2001). https://doi.org/10.1023/A:1010933404324
9. Cao, X., Tsang, I.: Bayesian active learning by disagreements: A geometric perspective. ArXiv **abs/2105.02543** (2021)
10. Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 785–794 (08 2016). https://doi.org/10.1145/2939672.2939785
11. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: Proceedings of the 37th International Conference on Machine Learning. pp. 1597–1607. ICML'20, JMLR.org (2020)
12. Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A., Padlewski, P., Salz, D., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., Kolesnikov, A., Puigcerver, J., Ding, N., Rong, K., Akbari, H., Mishra, G., Xue, L., Thapliyal, A., Bradbury, J., Kuo, W., Seyedhosseini, M., Jia, C., Ayan, B., Riquelme, C., Steiner, A., Angelova, A., Zhai, X., Houlsby, N., Soricut, R.: Pali: A jointly-scaled multilingual language-image model. In: Arxiv (2022)
13. Chicheng, Z., Kamalika, C.: Active learning from weak and strong labelers. In: Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 28. Curran Associates, Inc. (2015), https://proceedings.neurips.cc/paper/2015/file/eba0dc302bcd9a273f8bbb72be3a687b-Paper.pdf
14. Clark, K., Luong, M., Le, Q., Manning, C.: Electra: Pre-training text encoders as discriminators rather than generators. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net (2020), https://openreview.net/forum?id=r1xMH1BtvB
15. Clevert, D., Unterthiner, T., Hochreiter, S.: Fast and accurate deep network learning by exponential linear units(elus). In: Bengio, Y., LeCun, Y. (eds.) 4th International Conference on Learning Representations, ICLR 2016, Conference Track Proceedings (2016), http://arxiv.org/abs/1511.07289
16. Cordeiro, F., Carneiro, G.: A survey on deep learning with noisy labels: How to train your model when you cannot trust on the annotations? In: The 33rd SIBGRAPI Conference on Graphics, Patterns and Images. pp. 9–16 (11 2020). https://doi.org/10.1109/SIBGRAPI51738.2020.00010
17. C.Shui, F.Zhou, C.Gagn'e, B.Wang: Deep active learning: Unified and principled method for query and training. In: International Conference on Artificial Intelligence and Statistics (2020)
18. Du, P., Zhao, S., Chen, H., Chai, S., Chen, H., Li, C.: Contrastive coding for active learning under class distribution mismatch. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 8907–8916 (2021). https://doi.org/10.1109/ICCV48922.2021.00880
19. Eitan, A., Smolyansky, E., Harpaz, I., Perets, S.: Connected papers (2019), https://www.connectedpapers.com/
20. Fei-Fei, L., Andreetto, M., Ranzato, M., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories (2004). https://doi.org/10.22002/D1.20086
21. Gal, Y., Islam, R., Ghahramani, Z.: Deep bayesian active learning with image data. In: International Conference of Machine Learning. vol. abs/1703.02910, pp. 1183–1192 (2017)
22. Geifman, Y., El-Yaniv, R.: Deep active learning over the long tail. ArXiv **abs/1711.00941** (2017)
23. Gonog, L., Zhou, Y.: A review: Generative adversarial networks. In: The 14th IEEE Conference on Industrial Electronics and Applications (ICIEA). pp. 505–510 (2019). https://doi.org/10.1109/ICIEA.2019.8833686
24. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K. (eds.) Advances in Neural Information Processing Systems. vol. 27. Curran Associates, Inc. (2014), https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

25. Górriz, M., Carlier, A., Faure, E., i Nieto, X.G.: Cost-effective active learning for melanoma segmentation. ArXiv **abs/1711.09168** (2017)
26. Grave, E., Bojanowski, P., Gupta, P., Joulin, A., Mikolov, T.: Learning word vectors for 157 languages. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018) (2018)
27. Gupta, G., Sahu, A., Lin, W.: Noisy batch active learning with deterministic annealing. In: arXiv: Learning (2020)
28. Han, B., Yao, J., Niu, G., Zhou, M., Tsang, I., Zhang, Y., Sugiyama, M.: Masking: A new perspective of noisy supervision. In: Advances in Neural Information Processing Systems (05 2018)
29. Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., Sugiyama, M.: Co-teaching: Robust training of deep neural networks with extremely noisy labels. In: Advances in Neural Information Processing Systems (2018)
30. Har-Peled, S., Roth, D., Zimak, D.: Maximum margin coresets for active and noise tolerant learning. In: International Joint Conferences on Artificial Intelligence (2007)
31. Hataya, R., Nakayama, H.: Investigating cnns' learning representation under label noise. In: International Conference on Learning Representations (2018)
32. Haußmann, M., Hamprecht, F., Kandemir, M.: Deep active learning with adaptive acquisition. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. p. 2470–2476. IJCAI'19, AAAI Press (2019)
33. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016)
34. Hochreiter, S., Schmidhuber, J.: Long short-term memory. In: Neural Computation. vol. 9, pp. 1735–1780 (1997)
35. Huang, G., Liu, Z., Maaten, L.V.D., Weinberger., K.: Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2261–2269 (2017). https://doi.org/10.1109/CVPR.2017.243
36. Huang, T., Lihong, L., Vartanian, A., Amershi, S., Zhu, X.: Active learning with oracle epiphany. In: Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 29. Curran Associates, Inc. (2016), https://proceedings.neurips.cc/paper/2016/file/299fb2142d7de959380f91c01c3a293c-Paper.pdf
37. Ivakhnenko, G., Lapa, V.: The group method of data handling. Automation and remote control **26**(6), 895–902 (1965)
38. Jia, D., Wei, D., Richard, S., Li-Jia, L., Kai, L., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (06 2009). https://doi.org/10.1109/CVPR.2009.5206848
39. Kolesnikov, A., Dosovitskiy, A., Weissenborn, D., Heigold, G., Uszkoreit, J., Beyer, L., Minderer, M., Dehghani, M., Houlsby, N., Gelly, S., Unterthiner, T., Zhai, X.: An image is worth 16x16 words: Transformers for image recognition at scale. In: 9th International Conference on Learning Representations, ICLR 2021 (2021), https://openreview.net/forum?id=YicbFdNTTy
40. Konyushkova, K., Sznitman, R., Fua, P.: Learning active learning from data. In: NIPS (2017)
41. Krizhevsky, A., Nair, V., Hinton, G.: Cifar-100 (canadian institute for advanced research) (2009), http://www.cs.toronto.edu/~kriz/cifar.html
42. Lecun, Y., Bottou, L., Bengio., Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (09 1998). https://doi.org/10.1109/5.726791
43. LeCun, Y., Cortes, C., Burges, C.: Mnist handwritten digit database. ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist **2** (2010)
44. Lewis, D., Gale, W.: A sequential algorithm for training text classifiers. In: SIGIR '94. pp. 3–12. Springer London, London (1994)
45. Li, X., Yang, P., Wang, T., Zhan, X., Xu, M., Dou, D., Xu, C.: Deep active learning with noise stability. In: International Conference on Learning Representations (05 2022). https://doi.org/10.48550/arXiv.2205.13340
46. Lin, T., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Doll'a r, P., Zitnick, C.: Microsoft COCO: common objects in context. CoRR **abs/1405.0312** (2014), http://arxiv.org/abs/1405.0312
47. Liu, H., Cheng, G., Lin, W., Yang, J., Yang, J., Zhang, H., Zhang, Z., Wu, W.: Swin transformer: Hierarchical vision transformer using shifted windows. In: International Conference on Computer Vision (ICCV) (2021)
48. M. Tan, Q.L.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. pp. 6105–6114. PMLR (2019)
49. Malach, E., Shalev-Shwartz, S.: Decoupling "when to update" from "how to update". In: Proceedings of the 31st International Conference on Neural Information Processing Systems. p. 961–971. NIPS'17, Curran Associates Inc., Red Hook, NY, USA (2017)
50. McCallum, A., Nigam, K.: Employing em and pool-based active learning for text classification. In: International Conference of Machine Learning (1998)
51. Mescheder, L., Geiger, A., Nowozin, S.: Which training methods for gans do actually converge? In: The 35th International Conference on Machine Learning (2018)

52. Nagarajan, N., Inderjit, D., Pradeep, R., Ambuj, T.: Learning with noisy labels. In: Advances in Neural Information Processing Systems. vol. 26. Curran Associates, Inc. (2013), https://proceedings.neurips.cc/paper/2013/file/3871bd64012152bfb53fdf04b401193f-Paper.pdf
53. Nair, V., Hinton, G.: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on International Conference on Machine Learning. p. 807–814. ICML'10, Omnipress, Madison, WI, USA (2010)
54. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.: Reading digits in natural images with unsupervised feature learning. In: NIPS Workshop on Deep Learning and Unsupervised Feature Learning (2011)
55. Novák, A., Laki, L., Novák, B.: CBOW-tag: a modified CBOW algorithm for generating embedding models from annotated corpora. In: Proceedings of the Twelfth Language Resources and Evaluation Conference. pp. 4798–4801. European Language Resources Association (09 2020), https://aclanthology.org/2020.lrec-1.590
56. Patrinin, G., Rozza, A., Menon, A., Nock, R., Qu, L.: Making deep neural networks robust to label noise: A loss correction approach. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2233–2241 (07 2017). https://doi.org/10.1109/CVPR.2017.240
57. Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (10 2014). https://doi.org/10.3115/v1/D14-1162, http://www.aclweb.org/anthology/D14-1162
58. Phillips, J.: Coresets and sketches. CoRR **abs/1601.00617** (2016), http://arxiv.org/abs/1601.00617
59. Phillips, J., Tai, W.: Near-optimal coresets of kernel density estimates. In: 34th International Symposium on Computational Geometry, SoCG 2018, June 11-14, 2018, Budapest, Hungary. LIPIcs, vol. 99, pp. 66:1–66:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2018). https://doi.org/10.4230/LIPIcs.SoCG.2018.66, https://doi.org/10.4230/LIPIcs.SoCG.2018.66
60. Popov, S., Morozov, S., Babenko, A.: Neural oblivious decision ensembles for deep learning on tabular data. ArXiv **abs/1909.06312** (2020)
61. Prokhorenkova, L., Gleb, G., Vorobev, A., Dorogush, A., Gulin, A.: Catboost: unbiased boosting with categorical features. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Proceedings of the 32nd International Conference on Neural Information Processing Systems. vol. 31, p. 6639–6649. Curran Associates, Inc. (2018)
62. Ren, P., Xiao, Y., Chang, X., Huang, P., Li, Z., Chen, X., Wang, X.: A survey of deep active learning. ACM Computing Surveys (CSUR) **54**, 1 – 40 (2020)
63. Roman, L., Valeriia, C., Avi, S., Arpit, B., Bruss, C., Tom, G., Andrew, W., Micah, G.: Transfer learning with deep tabular models (06 2022). https://doi.org/10.48550/arXiv.2206.15306
64. Rosenblatt, F.: The perceptron: A probabilistic model for information storage and organization in the brain. In: Psychological Review. vol. 65, pp. 386–408 (1958)
65. Scale.ai: Scale ai (2022), https://scale.com/
66. Schäfl, B., Gruber, L., Bitto-Nemling, A., Hochreiter, S.: Hopular: Modern hopfield networks for tabular data. ArXiv **abs/2206.00664** (2022)
67. See, A., Liu, P., Manning, C.: Get to the point: Summarization with pointer-generator networks. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. vol. 1, pp. 1073–1083 (2017)
68. Sener, O., Savarese, S.: Active learning for convolutional neural networks: A core-set approach. International Conference on Learning Representations (Poster) (2018), http://dblp.uni-trier.de/db/conf/iclr/iclr2018.html#SenerS18
69. Settles, B.: Active learning literature survey. In: ECML PKDD Workshop on Active Learning and Experimental Design. pp. 11–46 (2009)
70. Shui, C., Zhou, F., Gagn'e, C., Wang, B.: Deep active learning: Unified and principled method for query and training. In: AISTATS (2020)
71. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. International Conference on Learning Representations **abs/1409.1556** (2015)
72. Sinha, K., Zhang, Y., Doshi, P., Dhillon, I.: Efficient active learning for deep neural networks. In: Proceedings of the 36th International Conference on Machine Learning. pp. 3560–3568 (2019)
73. Sinha, S., Ebrahimi, S., Darrell, T.: Variational adversarial active learning. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 5971–5980. IEEE Computer Society, Los Alamitos, CA, USA (03 2019). https://doi.org/10.1109/ICCV.2019.00607, https://doi.ieeecomputersociety.org/10.1109/ICCV.2019.00607
74. Srinivas1, A., Lin, T., Parmar, N., Shlens, J., Abbeel, P., Vaswani, A.: Bottleneck transformers for visual recognition. In: 2021 Conference on Computer Vision and Pattern Recognition (2021)
75. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2015)
76. Touvron, H., Caballero, J., Guillaumin, M., Jégou, H.: Going deeper with transformers: A study of deep multi-head attention models for image classification. In: International Conference on Computer Vision (ICCV) (2020)

77. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. vol. 30 (2017), https://arxiv.org/pdf/1706.03762.pdf
78. Wang, C., Singla, A., Chen, Y.: Teaching an active learner with contrastive examples. In: Advances in Neural Information Processing Systems (2021)
79. Wang, T., Li, X., Yang, P., Hu, G., Zeng, X., Huang, S., Xu, C., Xu, M.: Boosting active learning via improving test performance. In: AAAI Conference on Artificial Intelligence (2021)
80. Wei, J., Zhu, Z., Cheng, H., Liu, T., Niu, G., Liu, Y.: Learning with noisy labels revisited: A study using real-world human annotations. 10th International Conference on Learning Representations (2022)
81. Worker, C.: Clickworker (2022), https://www.clickworker.com
82. Wortsman, M., Ilharco, G., Gadre, S., Roelofs, R., Gontijo-Lopes, R., Morcos, A., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., Schmidt, L.: Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In: International Conference on Machine Learning. pp. 23965–23998. PMLR (2022)
83. Xia, X., Liu, T., Han, B., Wang, N., Gong, M., Liu, H., Niu, G., Tao, D., Sugiyama, M.: Part-dependent label noise: Towards instance-dependent label noise. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. NIPS'20, Curran Associates Inc., Red Hook, NY, USA (2020)
84. Xiaoqi, J., Yichun, Y., Lifeng, S., Xin, J., Xiao, C., Linlin, L., Fang, W., Qun, L.: TinyBERT: Distilling BERT for natural language understanding. In: Findings of the Association for Computational Linguistics: EMNLP 2020". pp. 4163–4174. Association for Computational Linguistics, Online (2020), https://aclanthology.org/2020.findings-emnlp.372
85. Yan, S., Chaudhuri, K., Javidi, T.: Active learning from imperfect labelers. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. p. 2136–2144. NIPS'16, Curran Associates Inc., Red Hook, NY, USA (2016)
86. Yi, J., Seo, M., Park, J., Choi, D.: Using self-supervised pretext tasks for active learning. In: European Conference on Computer Vision (2022)
87. Yoo, S., Kim, H., Kim, I., Kim, M., Hwang, S.: Active learning for deep neural networks. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 2070–2079. JMLR. org (2017)
88. Younesian, T., Epema, D., Chen, L.: Active learning for noisy data streams using weak and strong labelers. ArXiv **abs/2010.14149** (2020)
89. Younesian, T., Zhao, Z., Ghiassi, A., Birke, R., Chen, L.: Qactor: Active learning on noisy labels. In: Balasubramanian, V.N., Tsang, I. (eds.) Proceedings of The 13th Asian Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 157, pp. 548–563. PMLR (17–19 Nov 2021), https://proceedings.mlr.press/v157/younesian21a.html
90. Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., Wu, Y.: Coca: Contrastive captioners are image-text foundation models. Transactions on Machine Learning Research **abs/2205.01917** (2022)
91. Zihang, D., Hanxiao, L., Quoc, L., Mingxing, T.: Coatnet: Marrying convolution and attention for all data sizes. In: 35th Conference on Neural Information Processing Systems (06 2021)