An overview of Active Learning

Nam D. Tran
ERP Batch 1 - VinAI
Corresponding address
nam.td12@vinai.io

Abstract

The acquisition of a large number of high-quality annotated datasets for Deep Learning required a lot of resources. The fields that require high levels of expertise are impossible to run the annotated process. Therefore, Active Learning is become more popular and gets more attention from researchers. Deep Active Learning has emerged as the problem solver when it reduces the cost of sample annotations while retaining the powerful learning capabilities of Deep Learning. In this survey, the road map picture of this field is shown, and analyzing some high-quality papers have public recently.

1. Introduction

Deep Learning has a long period of development and proof that it can be the future of technology. By the advantages of gradients algorithms and the use of multiple GPUs effectively greatly. Deep Learning began to win championships in various competitions and constantly beat records in various tasks. Recently, many quality datasets have published, like in [1, 2] in image classification. But the manual labeling of datasets comes at a high cost, especially the fields with high degrees of professional knowledge such as medical data in [3], satellite data in [4]. The difficulties of labeling data in those fields come with the need of maximizing the performance gain of the model when required smaller data than normal model deep learning.
Active is the method that selects the most valuable data from the unlabeled dataset and hands it over to the oracle for labeling. The main idea is to reduce the cost of labeling as much as possible while trying to remain performance. There are three types of Active Learning approach divided into *pool-based* [5], *stream-based selective sampleing* [6] and *membership query synthesis* [6]. In the Stream-based scenario, the algorithm determines if it would be beneficial enough to query for the label of a specific unlabeled entry in the dataset. While the model is being trained, it is presented with a data instance and immediately decides if it wants to query the label. This approach has a natural disadvantage that comes from the lack of guarantee that the data scientist will stay within budget. In a Pool-based scenario, training examples are chosen from a large pool of unlabelled data. Selected training examples from this pool are labeled by the oracle. The downside of this method is the amount of memory it can require. In the membership query synthesis method, the active learner in this method is allowed to create its own examples for labeling. This method is compatible with problems where it is easy to generate a data instance. The Deep Active learning example process is shown in Figure 1. In detail, the process receives the L_0 training set to initialized the pre-trained Deep Learning model with parameters θ, the samples of the unlabeled pool U are used to extract features through the Deep Learning model. The query strategy is chosen to select sample data and query the label in querying to form a new label training set L, then the model Deep Learning is trained on L, and update U at the same time. The process is repeated until the label budget is exhausted or the conditions are reached.

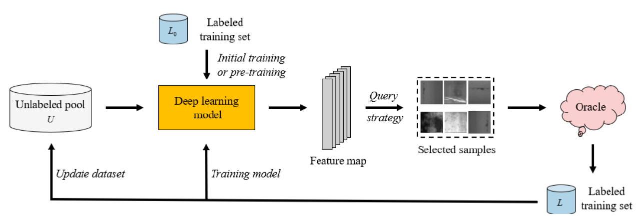

Figure 1. Example of a Deep Active Learning process

2. Deep Active Learning

Active learning has crucial potential to effectively reduce labeling costs, while Deep Learning has a strong learning ability in high-dimensional datasets. As a result, the combination of two methods is an obvious approach to accelerate AI Industrial. The combination leverage advantages of the two methods, but implement Active Learning in Deep Learning is hindered by the complexity of increasing the number of dimensions of the sample, and various difficulties. Such as: Insufficient data for label samples, the one-by-one data query method in Active Learning cannot apply for the Deep Learning training phase [7], this problem can be addressed by using generative networks for data augmentation [8]. Model uncertainty, one of the most important directions for Active Learning research community is based uncertainty. The Deep Learning model can use softmax as the final layer to clarify the probability of the label, but it can be too confident. Authors in [9] show method to handle the problem by trying to learn loss for active learning method, authors in [10] applied Bayesian deep learning to deal with the high-dimensional then reduce the problem that model being too confident about the output results. Another issue is processing pipeline inconsistency, most Active Learning frameworks based on fixed feature representations and focus on the training phase of classifiers, while Deep Learning is jointly optimized between feature learning and classifier training [11].

2.1. Query Strategy Optimization

The setup of DAL in pool-based method include unlabeled dataset $U^n = X, Y$ and the current training samples $L^m = X, Y$ (m samples). X, Y are the sample space and label space. The main target is to design a query strategy $Q : U^n \to L^m$, using deep model $f \in F, f : X \to Y$. As a result, the smaller m, the more accuracy of the model. The list below shows the most efficient strategy recently.

- Batch Mode DAL - uses batch-based sample querying
- Uncertainty-based and hybrid query strategies
- Deep Bayesian Active Learning
- Density-based Methods

2.2. DAL framwork

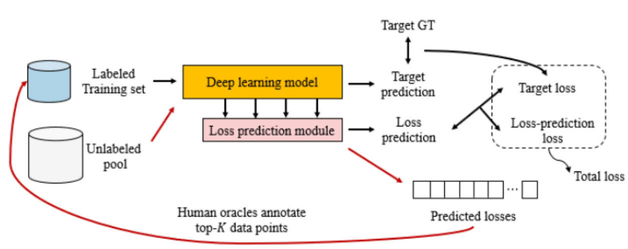

Figure 2. Example of a Deep Active Learning process

The inconsistency between Deep Learning and Active Learning need to resolve. One solution is only fine-tuning the Deep Learning model in the Active Learning framework, while the other simply combining Active Learning and Deep Learning to treat them as two separate problems, which may cause divergence. The Figure 2 shows the Example of a Deep Active Learning framework that can solve the problem address above. The black line represents the stage of training model parameters, optimizing the overall loss composed of target loss and loss-prediction loss. The red line represents the sample query phase of Active Learning. The output of the multiple hidden layers of the Deep Learning model is used as the input of the loss prediction module, while the $top - K$ unlabeled data points are selected according to the predicted losses and assigned labels by the oracle.
In the next section, three prestigious papers recently are shown to address the importance of Deep Active Learning.

3. Bayesian Generative Active Deep Learning

3.1. Proposed method

The Bayesian Generative Active Deep Learning is published in 2019 at ICML [8].
The authors introduce that there are two methods to address the data issue, data augmentation, and active learning. With the growth of enormous datasets, the data augmentation method can be costly and ineffective. The active learning method comes as the solver - instance is Pool-based active learning. Pool-based relied on the use of a small labeled data set and a large unlabeled data set, where small subsets from the unlabeled set are automatically selected using an acquisition function that assesses how informative those subsets are for the training process. These selected unlabeled subsets are then labeled by an oracle (i.e., a human annotator), integrated into the labeled data set, which is then used to re-train the model in an iterative training process. This method can be good at server situations but can overfit the informative training sets due to their small sizes. Additionally, the state-of-the-art model GAN [12] is a better way to generate data, but has a disadvantage. It uses the samples from the labeled set to guide the generation of new artificial training points by sampling from a generative distribution that is assumed to have a particular shape. Results on a large proportion of the generated samples will not be important for the training process.
To handle both issues of the above methods, the author introduces the Bayesian generative active deep learning method that targets the augmentation of the labeled data set with generated samples that are informative for the training process.
Contribution of the paper:

- Query by synthesis active learning.
- Bayesian data augmentation.
- Auxiliary-classifier generative adversarial networks (ACGAN) and (VAE).

The Bayesian active learning by disagreement (BALD) is the acquisition function in the active selection process. Using VAE-ACGAN to produce new artificial samples, new labels are then labeled by an oracle. New samples are then incorporated into the labeled data set to be used in the next training iteration.

3.2. Detail and relative

Bayesian Active Learning - The most important part of Pool-based Active Learning is the acquisition functions, there are some methods such as *expected informativeness* [13] and *expected error* of the learner [14]. But those methods are hard to implement in Deep Learning due to high-dimensional parameter vectors computation costs.
The idea using Monte Carlo (MC) dropout method in Bayesian active learning by disagreement [15] is shown to work well in practice despite the poor convergence of the MC approximation. With such an idea inherited, this article uses MC to approximate the BALD acquisition function in the active selection process.
Generative Active Learning - The authors introducing the Bayesian Generative Active Deep Learning model to generate informative data for the training phase. Firstly, the model queries the unlabeled data set samples based on their *information content*, and conditions the generation of a new synthetic sample on this selected sample. Both ACCGAN [16] and AE-GAN [17] are used in the Bayesian Generative Active Deep Learning framework to improving the classification performance.

3.3. Information-Preserving Data Augmentation for Active Learning

The main technical contribution of this paper consists of combining Bayesian Active Learning by Disagreement (BALD) and Bayesian data augmentation (BDA) for generating new labeled samples that are informative for the training process (see Figure. 3). The BDA is modified by conditioning the generation step on a sample x and a label y. The most informative sample x selected by $a(x, M)$ is s pushed to go through a variational autoencoder (VAE), , which contains an encoder $e(.)$ and a decoder $g(.)$, in order to generate the sample x' , as follows:

$$x' = g(e(x*)) \tag{1}$$

The label of x' is assumed to be y (i.e., the oracle's label for x) and the current labeled data set is then augmented with (x, y) and (x', y), which are used for the next training iteration. The informative content of the generated sample x' is proved by authors is informative [8].
Networks of framwork include: a classifier $c(x;_C)$, an encoder $e(x;_E)$, a decoder/generator $g(z;_G)$ and a discriminator $d(x;_D)$. The classifier $c(.)$ can be any modern deep convolutional neural network classifier [18]. Makeing the classifier can perform well in many tasks. The loss functions of VAE-GAN is:

$$L = L_{VAE} + L_{ACGAN}$$
$$L_{VAE} = L_{rec} + L_{prior}$$

Where L_{rec} and L_{prior} are the VAE loss [17].

4. Learning Loss for Active Learning

4.1. Proposed method

The Learning Loss for Active Learning is published in 2019 at CVPR [9].

The performance of recent methods semi-supervised learning and unsupervised learning [19, 20] but is still bound to that of fully-supervised learning. In real projects, the annotation phase for classification is relatively cheap but it contrasts with the bounding box. Then the active learning comes naturally.
The core idea of active learning is that the most informative data point would be more beneficial to model improvement than a randomly chosen data point. The authors try to make the active learning method work well on deep networks. The main idea is If we can predict the loss of a data point, it becomes possible to select data points that are expected to have high losses. The module is illustrated in Figure 4-(a). Once the module is learned, it can be utilized to active learning as shown in Figure 4-(b). The selected data points would be more informative to the current model. The *loss prediction module* is attached to the deep network and learns the module to predict the loss of an input data point.
The main approach is if the loss of a data point cannot be predicted, it becomes possible to select data points that are expected to have high losses. The selected data points would be more informative to the current model.
Contribution of the paper:

- Proposing loss prediction module of active learning method, which is directly applicable to any tasks with recent deep networks.
- Three learning tasks including classification, regression, and a hybrid of them are evaluated with the proposed method.

4.2. Detail and relative

The active learning scenario is defined with the proposal loss prediction module. The system is combined with model Θ_{target} and loss prediction module Θ_{loss} (Figure 4-(a)). Ouputs of deep network model is $\hat{y} = \Theta_{target}(x)$, and the loss of prediction module is $\hat{l} = \Theta_{loss}(h)$. Where h is a feature set of x extracted from several hidden layers of Θ_{target}. The unlabeled data pool is U_N and the process will take K data point at each state and denoted as L_K^{state}. The target Θ_{target}^{state} and loss prediction module $\Theta_{\substack{state \\ loss}}$ are learned

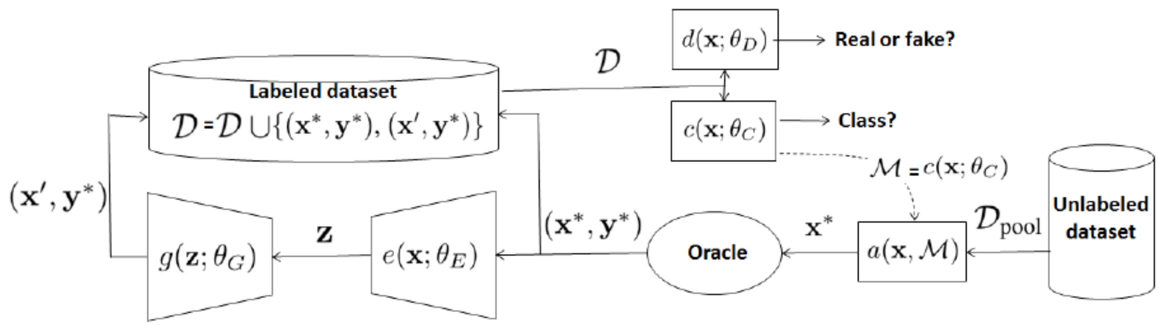

Figure 3. Network architecture of Bayesian Generative Active Deep Learning

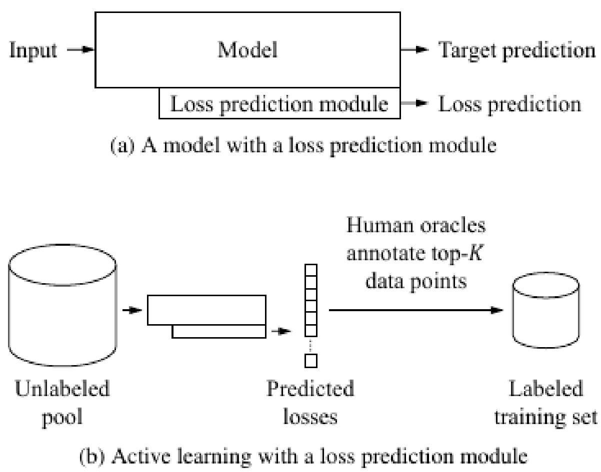

Figure 4. Active learning with a loss prediction module.

jointly. After training phase of each state, the data-loss pair is obtained in the unlabeled data pool $\{(x, \hat{l}) | x \in U^{state}_{N-K}\}$. Then, human oracles annotate the data points of the K -highest losses, illustrated in Figure 4-(b).

4.3. Loss prediction module

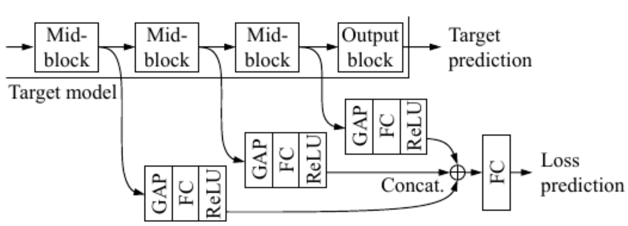

Figure 5. The architecture of the loss prediction module.

The loss prediction module is the main task of the active learning method since it learns to imitate the loss defined in the target model. This section describes how we design it. Figure 5 illustrates the architecture of our loss prediction module. It takes multi-layer feature maps h as inputs that are extracted between the mid-level blocks of the target model. These multiple connections let the loss prediction module choose necessary information between layers useful for loss prediction. Each feature map is reduced to a fixed dimensional feature vector through a global average pooling (GAP) layer and a fully-connected layer. Then, all features are concatenated and pass through another fully-connected layer, resulting in a scalar value $\hat{l}$ as a predicted loss. Learning this two-story module requires much less memory and computation than the target model. We have tried to make this module deeper and wider, but the performance does not change much. Figure 6 showed details of this module. In-state s, the labeled dataset is $L^s_{K.(s+1)}$ and the objective of this phase is to learn the model set $\{\Theta^s_{target}, \Theta^s_{loss}\}$. The target model $\hat{y} = \Theta_{target}(x)$ and the loss prediction module $\hat{l} = \Theta_{loss}(h)$ are obtained by training data point x. With the target annotation y of x, the target loss can be computed as $l = L_{target}(\hat{y}, y)$ to learn the target model. Since this loss l is a ground-truth target of h for the loss prediction module, the loss can be computed for the loss prediction module as $L_{loss}(l, l)$. Then, the final loss function to jointly learn both the target models and the loss prediction module is defined as the equation below, where λ is a scaling factor, and it usually constant.

$$L_{target}(\hat{y}, y) + \lambda L_{loss}(\hat{l}, l) \tag{2}$$

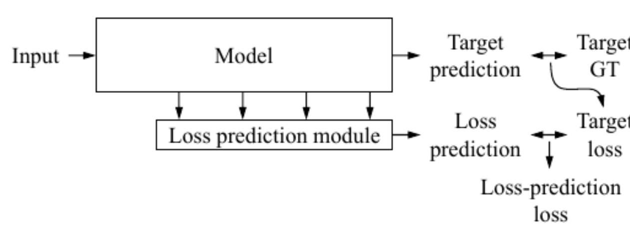

Figure 6. Method to learn the loss.

5. State-Relabeling Adversarial Active Learning

5.1. Proposed method

The State-Relabeling Adversarial Active Learning is published in 2020 at CVPR [21].
The state relabeling adversarial active learning model (SRAAL) leverages both the annotation and the labeled/unlabeled state information for deriving the most informative unlabeled samples. The generator tries to generate the unified representation of samples, which embeds the semantic into the whole data representation. The discriminator is designed with an online uncertainty indicator, which endues unlabeled samples with different importance. As a result, we can select the most informative samples based on the discriminator's predicted state. And the other contribution is to consider a realistic setting for pool-based semi-supervised AL. The active learning cycle showed in figure 8, set of samples is selected to be labeled from an unlabeled pool at each iteration. These selected unlabeled subsets are labeled by an oracle, integrated into the labeled data pool. How to select the most informative samples from the unlabeled pool is the key problem in active learning. State relabeling adversarial active learning model (SRAAL) that considers both the annotation and the state information for deriving most informative unlabeled samples. The generator is embedded information built by an unsupervised image reconstructor based on VAE architecture, and a supervised target learner to predict annotations for labeled samples, which endues the unlabeled samples with different importance.
The initialization of the labeled pool has a large influence on subsequent sample selection and performance of active learning. The k-center [22] approach is introduced to initialize the labeled pool, where selected samples are a diverse cover for the dataset under the minimax distance.
Contribution of the paper:

- State relabeling adversarial active learning model to select most informative unlabeled samples.
- Introduce an unsupervised image reconstructor and a supervised target learner to generate a unified representation of image are built.
- the initially sampling algorithm based on the $k-center$ approach

5.2. Detail and relative

State relabeling adversarial active learning model (SRAAL) is showed in Figure 7, which uses the annotation and labeled/unlabeled state information for selecting the most informative samples. The SRAAL consists of a unified representation generator and a labeled/unlabeled state discriminator. The former learns the annotation-embedded image feature, and the latter selects more representative samples to be labeled with the help of the online uncertainty indicator. A sampling strategy based on the generator and discriminator is introduced in Algorithm 1, and the proposed initial sampling algorithm with a $k-center$.
It consists of a unified representation generator and a labeled/unlabeled state discriminator. The generator embeds the annotation information into the final image features via the supervised target learner and unsupervised image reconstructor. Online uncertainty indicator is introduced to relabel the state of unlabeled samples and endues them with different importance. Finally, the state discriminator is updated through the labeled and unlabeled state losses and helps select the more informative samples.
The image representation learning is in the charge of the unified representation generator which consists of the unsupervised image reconstructor (UIR) and the supervised target learner (STL). The image encoder consists of a CNN and two FC modules. The CNN extracts the image features and then FC individually learns the two latent variables for STL and UIR. The UIR module is a variational autoencoder (VAE) in which a low dimensional latent space is learned based on a Gaussian prior. As this process does not require annotations and the reconstruction target is the image itself, samples from both the labeled pool and unlabeled pool contribute to this module. To embed the annotation information into the representation, we build a supervised target learner to predict annotations of the samples based on the representation in latent space. The STL is also a VAE network and its decoder does not decode the representation for reconstruction. The decoder of STL varies with different tasks. Therefore, the two above representations are concatenated together as the unified image representation.
To make full use of the state information, we introduce the adversarial learning into SRAAL, where a discriminator is built to model the state of samples. The previous works utilize the binary state information, where the state of unlabeled samples is set to 1 and that of labeled samples is set to 0. In fact, different samples in unlabeled pool have different contribution for target task, and an unlabeled sample has lower priority to be labeled if it is more similar to samples in labeled pool. To better use the state information, we propose the online uncertainty indicator (OUI) to calculate an uncertainty score to relabel the state of unlabeled data. The uncertainty score measures the distribution concentration of the unlabeled data and is bound to [0,1]. After the state relabeling, the state of unlabeled samples changes from the fixed binary label 1 to a new continuous state.
The algorithm for training the SRAAL at each iteration is shown in Figure 7. In the sampling step, the generator generates the unified representation for each unlabeled

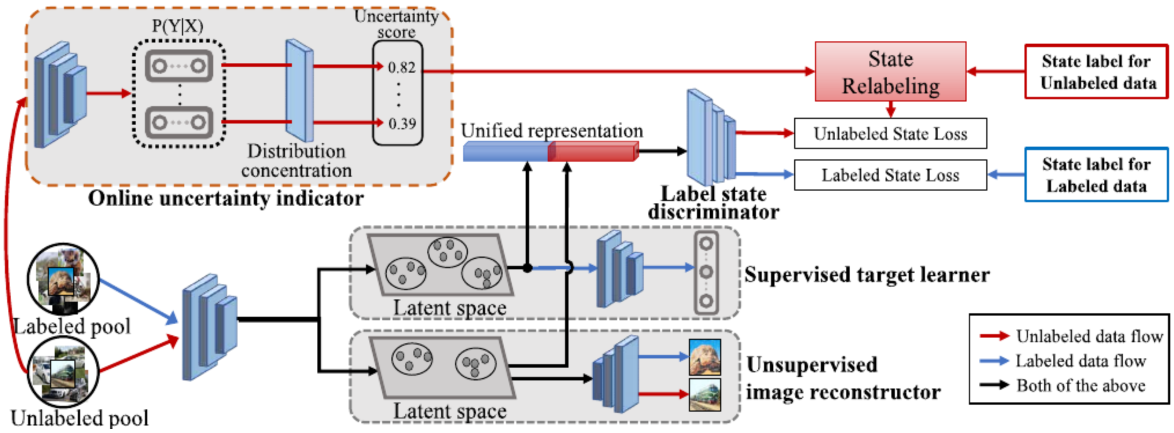

Figure 7. Network architecture of our proposed SRAAL

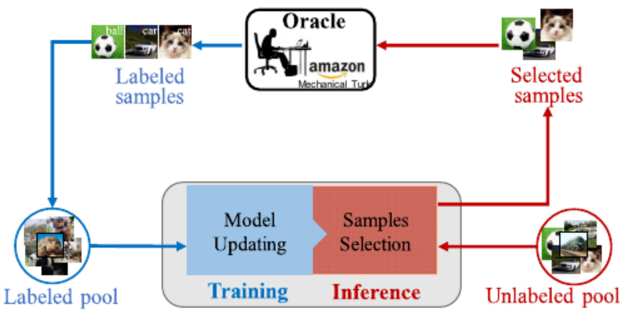

Figure 8. A traditional pool-based active learning cycle.

sample. The discriminator predicts its state value, and the $top-K$ samples are selected to be labeled by the oracle.

Data: Labeled data pool D_L , unlabeled pool D_U, initial labeled pool M (size), latent variables z for all the data points; Hyperparms - $I(I << M) \in D_U \rightarrow D_L$
Result: initialized labeled pool D_L
while $size(D_L) = M$ **do**
  $u = argmax_{x_U \in D_U}[min_{x_L \in D_L} Dist(x_U, x_L)]$
  $D_L = D_L U\{u\}$
  $D_U = D_U\{u\}$
end

Algorithm 1: Initialization of labeled pool

6. Conclusion

Active Learning, especially Deep Active Learning is an interesting field of the AI industry. The methods that DAL provides can bring great advantages when applying Deep Learning models in practice when the resources for the annotation are enormous. Deep Learning algorithms will be increasingly optimized and integrated to make the release process most profitable. The most recent work reveals that DAL has been successful in many common tasks. DAL has attracted the interest of a large number of researchers by reducing the cost of annotation and its ability to implement the powerful feature extraction capabilities of Deep Learning. Consequently, the related research work is also extremely rich. However, there are still a large number of unanswered questions on this subject. The current research directions regarding DAL methods focus primarily on the improvement of AL selection strategies, the optimization of training methods, and the improvement of task-independent models. Obviously, there is no conflict between the above-mentioned improvement directions. Thus, a mixed improvement strategy is an important development direction for the future. In general, DAL research has significant practical application value in terms of both labeling costs and application scenarios. However, DAL research remains in its infancy at present, and there is still a long way to go in the future.

Many thankfully for the VInAI cause of sponsored this paper. Directly contacted or contributed questions can be reached via the link: `https://www.vinai.io/careers/`.

References

[1] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115 (3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[2] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[3] Samuel Budd, Emma C Robinson, and Bernhard Kainz.

A survey on active learning and human-in-the-loop deep learning for medical image analysis. *arXiv preprint arXiv:1910.02923*, 2019.

[4] Peng Liu, Hui Zhang, and Kie B Eom. Active deep learning for classification of hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(2):712–724, 2016.

[5] Andrew Kachites McCallumzy and Kamal Nigamy. Employing em and pool-based active learning for text classification. In *Proc. International Conference on Machine Learning (ICML)*, pages 359–367. Citeseer, 1998.

[6] Burr Settles. Active learning literature survey. 2009.

[7] Fedor Zhdanov. Diverse mini-batch active learning. *arXiv preprint arXiv:1901.05954*, 2019.

[8] Toan Tran, Thanh-Toan Do, Ian Reid, and Gustavo Carneiro. Bayesian generative active deep learning. In *International Conference on Machine Learning*, pages 6295–6304. PMLR, 2019.

[9] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 93–102, 2019.

[10] Yarin Gal and Zoubin Ghahramani. Bayesian convolutional neural networks with bernoulli approximate variational inference. *arXiv preprint arXiv:1506.02158*, 2015.

[11] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600, 2016.

[12] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

[13] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

[14] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996.

[15] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International Conference on Machine Learning*, pages 1183–1192. PMLR, 2017.

[16] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*, pages 2642–2651. PMLR, 2017.

[17] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[18] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[19] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *Proceedings of the IEEE international conference on computer vision*, pages 37–45, 2015.

[20] Antti Rasmus, Harri Valpola, Mikko Honkala, Mathias Berglund, and Tapani Raiko. Semi-supervised learning with ladder networks. *arXiv preprint arXiv:1507.02672*, 2015.

[21] Beichen Zhang, Liang Li, Shijie Yang, Shuhui Wang, Zheng-Jun Zha, and Qingming Huang. State-relabeling adversarial active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8756–8765, 2020.

[22] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.