PAL : Pretext-based Active Learning

Shubhang Bhatnagar[1]
160020019@iitb.ac.in
Sachin Goyal[2]
sachingo@andrew.cmu.edu
Darshan Tank[1]
darshantank3@gmail.com
Amit Sethi[1]
asethi@iitb.ac.in

[1] Indian Institute of Technology, Bombay
[2] Carnegie Mellon University

Abstract

The goal of pool-based active learning is to judiciously select a fixed-sized subset of unlabeled samples from a pool to query an oracle for their labels, in order to maximize the accuracy of a supervised learner. However, the unsaid requirement that the oracle should always assign correct labels is unreasonable for most situations. We propose an active learning technique for deep neural networks that is more robust to mislabeling than the previously proposed techniques. Previous techniques rely on the task network itself to estimate the novelty of the unlabeled samples, but learning the task (generalization) and selecting samples (out-of-distribution detection) can be conflicting goals. We use a separate network to score the unlabeled samples for selection. The scoring network relies on self-supervision for modeling the distribution of the labeled samples to reduce the dependency on potentially noisy labels. To counter the paucity of data, we also deploy another head on the scoring network for regularization via multi-task learning and use an unusual self-balancing hybrid scoring function. Furthermore, we divide each query into sub-queries before labeling to ensure that the query has diverse samples. In addition to having a higher tolerance to mislabeling of samples by the oracle, the resultant technique also produces competitive accuracy in the absence of label noise. The technique also handles the introduction of new classes on-the-fly well by temporarily increasing the sampling rate of these classes. We make our code publicly available at https://github.com/shubhangb97/PAL_pretext_based_active_learning

1 Introduction

In spite of their unprecedented accuracy on several tasks involving image analysis, a hurdle in using convolutional neural networks (CNNs) for many real problems is their requirement of large labeled datasets. Labeling and annotations are laborious and costly for several domains, such as medical imaging, where expertise or follow-up is required. Strategies to reduce the number of labels include transfer, semi-supervised, few-shot, and active learning. Active learning algorithms are used to decide whether or not to send an unlabeled sample for labeling to an oracle (e.g., a radiologist for x-ray images), such that the increase in the task performance (e.g., classification accuracy) is maximized with respect to a labeling cost.

In pool-based active learning, training progresses iteratively in rounds or queries starting from an unlabeled pool of samples. In each query, up to a budgeted number of N additional samples can be selected from the unlabeled pool for labeling [8, 27, 29]. After a random initial selection of samples for labeling, the query selection strategy is usually based on picking *novel* and *diverse* samples from the unlabeled pool. Novelty (a.k.a. *uncertainty* and *confusion*) refers to selecting samples that are least similar to the previously labeled samples in order to maximize the information gain by getting them labeled. The samples in a query should also be diverse in order to maximize the collective information gained by their labeling.

While oracles are assumed to be ideal, in a realistic scenario we cannot expect the oracle to label all samples correctly. Therefore, we need to reduce the dependence of the query selection on the labels as well as on the task network (CNN that is trained to perform the task, e.g., classification) that models the distribution of the labeled data. Secondly, it recently became clear that the task network is a poor estimator of its own uncertainty on unlabeled samples that are unlike the labeled samples [16]. While we also model the distribution of the labeled data in order to select the samples for a query, our main contribution is to do so using a self-supervised learning (SSL) task (Section 3.1) that reduces the dependence on potentially noisy labels. Additionally, we perform the query selection using a second network – the *scoring network* – in order to exercise greater control over query formation [32], unlike most of the previously proposed active learning methods that rely on the task network itself for estimating uncertainty [2, 8, 10, 29].

In other words, we use the extent to which an unlabeled sample gives a wrong prediction on the SSL task as an indicator of its novelty. The self-supervision labels can be generated inexpensively for testing the uncertainty of the unlabeled samples, compared to several other techniques. Due to the use of the pretext (SSL) task, we call this scheme *pretext-based active learning (PAL)*.

Additionally, we regularize the scoring network to counter the paucity of labeled data by adding a classification head for multi-task learning. Finally, we ensure diversity among the N samples selected during a query by breaking it into K sub-queries. For each sub-query, we pick samples that are novel with respect to the previous sub-queries, which ensures diversity among the samples of the query itself. We tune only the self-supervision head between the sub-queries, so that we do not incur the labeling cost until the entire query is formulated (Section 3.3).

Due to its reliance on self-supervised learning, PAL seems to be significantly more robust to partial mislabeling of the training data by the oracle (Section 4.2). PAL also showed an accuracy that is competitive with the state-of-the-art [10, 27, 29] on benchmark image recognition and segmentation datasets, without using a computationally expensive training scheme (Section 4.1). We also tested PAL for a scenario called *biased initial pool*, in which certain classes may be underrepresented (or absent) in the initially labeled data. As desired, PAL over-samples the previously underrepresented classes and ramps up the performance on them in the first few queries itself, and then returns to balanced sampling (Section 4.3).

2 Related Work

2.1 Active learning

There are several settings for active learning, such as *membership query synthesis* and *stream-based sampling*. In the former, the learner generates new samples to query the oracle [1, 19, 34], while in the latter the unlabeled dataset is presented as a stream, and is evaluated online [6, 7]. However, unlike these settings, the proposed method is *pool-based sampling*, which makes a complete use of labeled and unlabeled data pools, when the latter is also available [10, 27, 29]. In this setting, starting with a set of labeled samples, a budgeted number of unlabeled samples are selected for querying the oracle.

Pool-based active learning techniques aim to pick samples that are novel and diverse. **Novelty** (a.k.a., uncertainty, confusion, non-triviality, out-of-distribution, and informativeness) refers to an unlabeled sample's ability to provide new information, if labeled, *independently of other samples selected*. Some of the early measures of novelty have known issues. For example, entropy of the estimated class probability mass function [28] is prone to calibration error [16], discordance between a committee of classifiers [12] can be computationally expensive, and distance from a linear decision boundary [30] is not directly applicable to CNNs because of their complex decision boundaries. Distance from an adversarial example has been proposed as an approximation of distance from decision boundary [8], but it requires computationally expensive gradient descent on image pixels. Surprisingly, no one has used the difficulty of solving a self-supervised (pretext) task as a measure of novelty, which requires only up to one additional network to be trained in parallel with the task network.

Methods based on **diversity** (a.k.a. representativeness and coverage) seek to select samples that can represent the unlabeled data distribution well. If the samples selected in a query are individually novel with respect to the previously labeled samples but collectively similar to each other, then the joint information gained from their labels as a query group may not be maximized. A method based on identifying a *core-set* has been proposed that models the empirical loss over the set of labeled samples on the empirical loss over the whole dataset [27]. However, this approach suffers when the representations are high-dimensional, because the Euclidean distance is a poor local similarity estimator in high dimensional spaces. An alternative approach called variational adversarial active learning (*VAAL*) aims to learn a good representation using a variational autoencoder (VAE) trained adversarially using a discriminator that tries to predict if a sample is already labeled [29]. However, this is also a computationally expensive technique due to VAE training.

2.2 Self-supervised learning

Self-supervised learning (SSL) has shown great promise in learning usable data representations without needing explicit data labels. Many SSL techniques automatically create a supervised pretext task by degrading an unlabeled image, and train a neural network to recover the original image. Some commonly used randomized degradations on an image for SSL are removing color [21], reducing resolution [22], occluding parts of an image [25], jumbling the spatial order of its sub-images [24], and applying random geometric transforms [11]. Several other recent SSL techniques are based on contrastive learning, like SimCLR[4] and MoCo[15] instead focus on making a CNN learn image representations that are closer for augmented versions of the same images compared to those of the others.

If a CNN trained using an SSL task can correctly solve the SSL puzzle on a test image, it can be interpreted that the test image is similar to some of the training images [13, 18]. Since training the scoring network using SSL on labeled samples does not require the oracle's labels, which may be noisy, we use the difficulty of solving the SSL task as a robust measure of the novelty of unlabeled images.

3 Method: Pretext-based Active Learning

Our method is an instance of pool-based active learning, essence of which can be described as follows. Let the pool of the currently labeled samples be $\mathcal{D}_L$ and the pool of unlabeled samples be $\mathcal{D}_U$. A task network $f_\theta(\mathbf{x}_l)$ parameterized by θ is trained on all samples $\mathbf{x}_l \in \mathcal{D}_L$. The active learning algorithm selects a budgeted set of N or fewer samples from $\mathcal{D}_U$ in each query. The queried samples are then labeled by an oracle (assumed ideal, although unrealistic), added to $\mathcal{D}_L$, and removed from $\mathcal{D}_U$. The task network is retrained on the expanded $\mathcal{D}_L$ and its increase in accuracy is examined. This process is repeated until a specified number of samples $|\mathcal{D}_L|$ are labeled or a desired accuracy level is achieved.

We use a different neural network than task network for our selection strategy, which we refer to as the *scoring network* hereafter. The scoring network has two heads, one for self-supervision and another for classification, whose outputs are used to assign a confusion score S to an unlabeled image $\mathbf{x}_u$.

3.1 Self-supervision head

The self-supervision head of the scoring network estimates the likelihood of the unlabeled data under the distribution of the labeled samples. We quantify this likelihood using the self-supervision score S_S, which we formulate for two self-supervised techniques, namely the prediction of randomized rotation for classification (SSL)[11], and SimCLR based on contrastive learning [4].

In the rotation task, we rotate the images by $90i^\circ$ for $i \in \{0,1,2,3\}$ and train a network g_ϕ parameterized by ϕ to predict i on only the images from $\mathcal{D}_L$, so that the head learns the distribution of the labeled data. Using the self-supervised head, the following confusion score S_S is assigned to each unlabeled image $\mathbf{x}_u$:

$$S_S(\mathbf{x}_u) = -\sum_{i\varepsilon\{0,1,2,3\}} g_\phi(\mathrm{rot}_{90i}(\mathbf{x}_u))_i, \tag{1}$$

where $\mathrm{rot}_{90i}(.)$ is the rotation function and $g_\phi(.)_i$ is the estimated probability of the i^{th} rotation angle. We hypothesize that an image $\mathbf{x}_u \in \mathcal{D}_U$ for which S_S is closer to its minimum possible value of -4 will likely be similar to the labeled points in $\mathcal{D}_L$, and will fetch little extra information, if labeled. Conversely, for novel points S_S will be closer to 0.

In SimCLR [4], random transformations, including crops, resizes, and color jitter, are applied to the images. The network g_ϕ is then trained to reduce the distance between embeddings of two such transformed versions of the same image. We train the network g_ϕ in this manner only on images belonging to $\mathcal{D}_L$, making the network learn the distribution of the labeled data. Specifically, for the semantic segmentation task, we use the embedding obtained from the bottleneck layer of the segmentation architecture to perform our contrastive learning task. Using the self-supervised head, we define the confusion score S_S, assigned to each unlabeled image $\mathbf{x}_u$ as:

$$S_S(\mathbf{x}_u) = sim(g_\phi(\mathbf{x}_{u,1}), g_\phi(\mathbf{x}_{u,2})) \tag{2}$$

where $\mathbf{x}_{u,1}$ and $\mathbf{x}_{u,2}$ are transformed versions of $\mathbf{x}_u$, and $sim(a,b)$ denotes the cosine similarity between 2 vectors a and b. An image $\mathbf{x}_u \in \mathcal{D}_U$ which has S_S closer to 1 will most likely be similar to points in $\mathcal{D}_L$, and will fetch less extra information. Other self-supervised tasks can also be used to generate a suitable estimate S_S for each unlabeled data image. In our experiments, we use the score defined in equation 1 for sample selection when the task is to perform classification, and the score in equation 2 when the task is to perform semantic segmentation.

3.2 Classification head and hybrid score

A scoring network trained only using the self-supervised pretext task might still be unable to model the labeled data distribution accurately. Its performance can be further augmented by the addition of a classification head to it, which is trained using the already available (though possibly noisy) labels of $\mathcal{D}_L$. It would help overcome the limitation of having a small initially labeled dataset by helping regularize the network. It would also help mitigate the unreliability of the score S_S for certain kinds of images, for example images having rotational symmetry. So, we introduce a classification head $h_\psi(\mathbf{x}_u)$ parameterized by ψ in the scoring network. We compute the degree to which the outputs of h_ψ for an unlabeled sample $\mathbf{x}_u$ are close to a uniform distribution U, using KL divergence (or relative entropy) [17], as a second measure of confusion $S_C(\mathbf{x}_u)$, to give a *hybrid confusion score* $S(\mathbf{x}_u)$, as shown below:

$$\begin{aligned} S(\mathbf{x}_u) &= S_S(\mathbf{x}_u) + \lambda S_C(\mathbf{x}_u), \text{ where} \\ S_C(\mathbf{x}_u) &= -\mathrm{KL}(U \,||\, h_\psi(\mathbf{x}_u)), \end{aligned} \tag{3}$$

where $\lambda \geq 0$ is a relative importance hyperparameter. When applying PAL to semantic segmentation, we calculate S_C pixel-wise and average it to get S_C for the sample.

Although, the entropy of class probabilities is a more popular measure of confusion [28], its range is finite. Had it been used as S_C in place of the negative of KL divergence in Equation 3 it would not have been able to counter-balance the effect of S_S (Equation 1) when it fails (e.g., in case of rotational symmetry). On the other hand, when the class prediction by h_ψ is very confident and KL-divergence is high, that means, as desired,we would rely less on the SSL task, and combined score will self-adjust due to the infinite range of the KL-divergence.

An added advantage of using a multi-task setting for the scoring network is getting better ordinal estimates of a true latent score due to an ensemble-like effect. We select the N most informative samples from $\mathcal{D}_U$ with the highest $S(\mathbf{x}_u)$ as per Equation 3 in each query round, after finding a good setting for the hyperparameter $\lambda \geq 0$ based on validation.

3.3 Diversity score

To ensure that the N samples in a query are diverse to cover the unlabeled data distribution, we divide the query into K sub-queries with $\frac{N}{K}$ samples each. For selecting the first sub-query, we select the top $\frac{N}{K}$ samples using the confusion score S from Equation 3. For the next sub-query, we fine-tune the scoring network on the samples from previous sub-query, in a self-supervised manner without asking the oracle for the labels in the middle of the main query. This fine-tuned network is then used to generate a score S_D, which is the same as the S_S defined in Equation 1, but with $g_{\phi'}$ in place of g_ϕ. Here, $g_{\phi'}$ are the parameters of the self-supervision head of the scoring network g_ϕ after fine-tuning. S_D promotes diversity as it would be small for data points which are similar to the points already selected in the previous sub-queries.

Now, we define an updated score S:

$$S(\mathbf{x}_u) = S_S(\mathbf{x}_u) + \lambda_1 S_C(\mathbf{x}_u) + \lambda_2 S_D(\mathbf{x}_u) \tag{4}$$

where $S_S(\mathbf{x}_u)$ and $S_C(\mathbf{x}_u)$ are the previously defined confusion score components as in equation 3 and are calculated for all unlabeled samples using g_ϕ, the scoring network before fine-tuning. Using Equation 4 we select another sub-query of $\frac{N}{K}$ samples, and the process repeats until we have N samples.

The process of selecting the query samples $\mathcal{D}_Q$ for one query round is described in Algorithm Box 1 (refer Appendix A) dubbed *pretext-based active learning (PAL)*. In each query round, the scoring network is trained using the cross entropy loss and the corresponding loss for the self-supervised technique in a multi-task framework, while the task network is trained using the cross-entropy loss. While $g_{\phi'}$ is fine-tuned during a sub-query, all networks are trained from scratch using the cross entropy loss $\mathcal{L}$ only after the oracle labels $\mathcal{D}_Q$.

4 Experiments and Results

We performed experiments on four datasets: (1) SVHN [23] : 10 class classification task on color images of house number digits (2) CIFAR-10 [20] : 10 class classification task on color images (3) Caltech-101 [9] : 101 class classification on color images and (4) Cityscapes [5] where semantic segmentation has to be performed on images of size 2048 × 1024, with each pixel needing to be classified into one of 19 classes.

We compared the performance of our approach with the following active learning strategies. (1) *Random sampling:* This is the simplest but nevertheless a strong baseline involving randomly picking samples to be labeled. (2) *Entropy:* This is a classical method where the sample uncertainty is modeled as the entropy of its predicted class probabilities. (3) *VAAL:* This technique uses a VAE to learn a feature space and then adversarially trains a discriminator on it [29]. (4) *DBAL:* This method uses Bayesian CNNs to estimate uncertainty (novelty) of unlabeled points [10]. (5) *Core-set:* This is a representation-based method for selecting the samples most different than the labeled samples and seeks to maximize the diversity of the samples to be picked for labeling [27].

Comparison between various active learning techniques was performed using a common experimental schema, in line with prior works [27, 29]. All techniques were used to iteratively expand the labeled dataset for training a common classifier architecture – VGG16 [33] or a common semantic segmentation architecture- Deeplabv3 [3] with a MobileNetv2 [26] backbone – from scratch during each query round. For the scoring network of the proposed PAL approach, we used a ResNet-18 [14] architecture. The average accuracy of five random initializations were computed. The initial labeled pool of samples was shared by all techniques. For the image classification datasets, the initial labeled pool comprised 10% of the whole dataset, and each query round added an additional 5% of the samples selected by the individual active learning technique. For semantic segmentation, the initial labeled pool consisted of 5% of the total dataset, each query round added an additional 1% samples to the labeled dataset, and mean intersection over union (mIoU) was used as the performance metric.

4.1 Performance with error-free labels

Figure 1 compares the mean performance over five random initializations of different techniques for different fractions of the data labeled. Our PAL strategy outperformed random sampling by a wide margin and seems to consistently outperform VAAL [29], DBAL [10], and core-set [27]. For instance, PAL requires only 20% of labeled SVHN images to achieve performance equal to that achieved by VAAL and DBAL using 30% labels, or a potential savings of 33% labels. Additionally, PAL requires only about 2 hours per query round to train on a single 11GB GPU for SVHN, whereas more computationally expensive methods such as VAAL [29] take more than 24 hours for the same. Out of the techniques compared only core-set [27] was faster than PAL, but its relative accuracy was quite variable across the datasets. Similar trends can be observed for semantic segmentation on CityScapes.

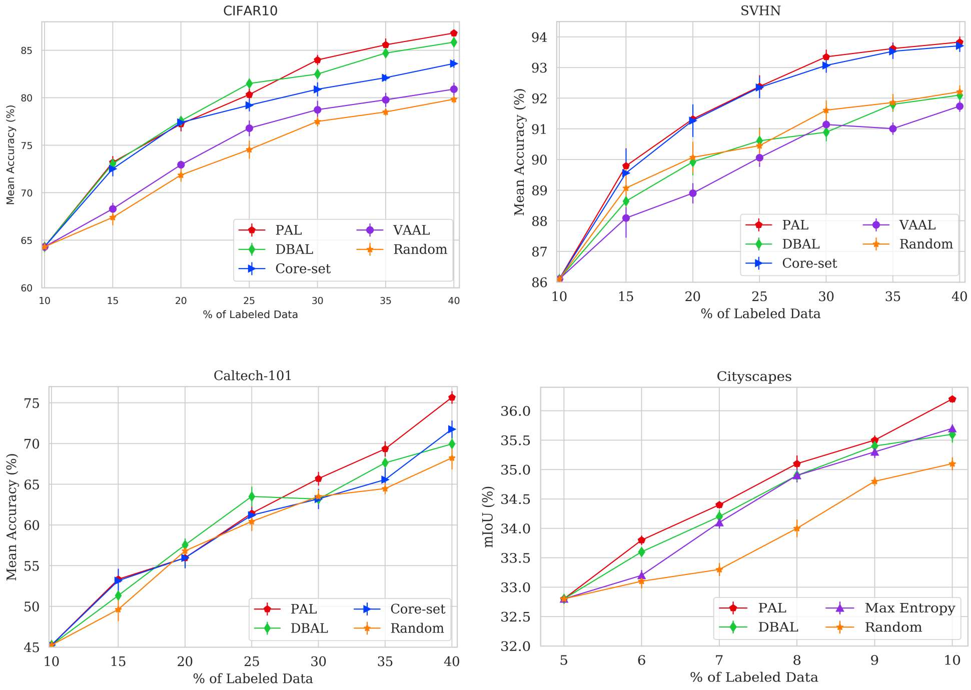

Figure 1: Performance of random sampling, entropy, VAAL [29], DBAL [10], and core-set [27] compared with PAL (proposed) on CIFAR-10, SVHN, Caltech-101, and Cityscapes (segmentation). Markers show mean accuracy of five runs, and vertical bars show standard deviation (some are too small to be visible). **Note that VAAL takes prohibitively long to train due to the use of a VAE. Therefore, we did not train VAAL on Caltech-101 and CityScapes.*

4.2 Robustness to sample mislabeling

We simulated labeling errors for classification by randomly assigning incorrect labels to a subset of the labeled pool and the queried set. We performed experiments on the SVHN and CIFAR-10 datasets, corrupting 20% of the data labels. In Figure 2, we observe that our technique clearly fares better compared to the others tested. We attribute this robustness of PAL to the use of the pretext task in the scoring network.

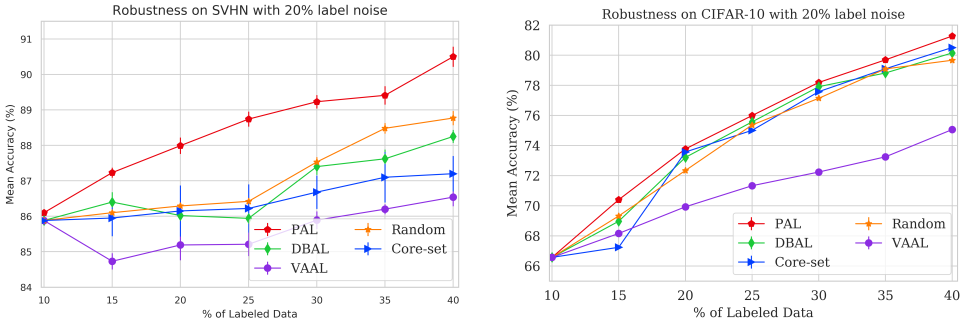

Figure 2: Active learning techniques compared with 20% label noise on SVHN (left) and CIFAR-10 (right).Markers show mean accuracy of five runs, and vertical bars show standard deviation.

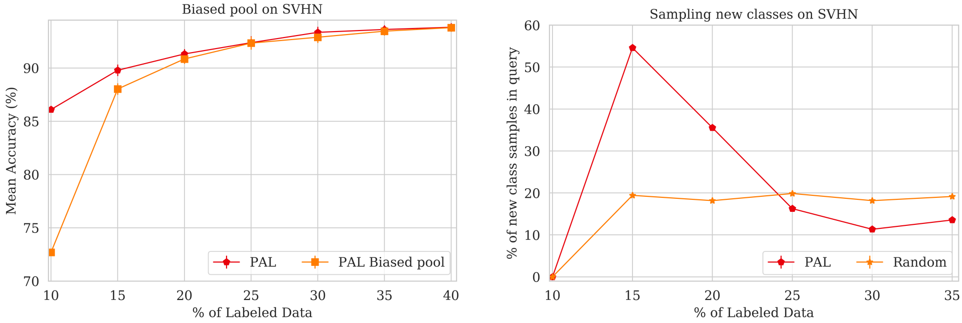

Figure 3: PAL performance with biased initial pool of only eight out of ten classes: The accuracy of PAL trained with biased pool quickly catches up with that of the trained without the biased initial pool (left), because it temporarily oversamples the newly introduced two classes that it finds novel but the random sampling does not (right).

4.3 Introducing new classes on-the-fly

We performed experiments with a biased initial pool consisting of only eight out of the ten classes in the SVHN dataset. After the initial training, the algorithm was given access to unlabeled samples from all the ten classes to check its behavior. As seen in Figure 3, PAL rapidly ramped up the performance when it was allowed to sample from the previously missing classes after the initial 10% labels. In fact, it quickly caught up with its own strong performance on the unbiased initial pool case (i.e., the upper-left graph of Figure 3 is same as that of SVHN results in Figure 1). PAL was able to temporarily over-sample the previously missing classes. On the other hand, the representation of the two missing classes remains around 20% for random sampling, once those classes are made available for queries, as expected. We observed similar trends for semantic segmentation on the Cityscapes dataset, where we started off with 17 out of the 19 classes and observed that PAL was able to select images which had upto seven times higher pixel area corresponding to missing classes compared to random (Figure 4).

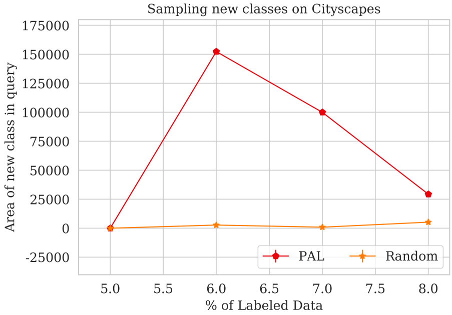

Figure 4: On CityScapes semantic segmentation dataset, we start with 17 classes out of 19 classes initially. PAL (our method) is able to sample images with upto seven times higher pixel areas of the missing classes compared to random

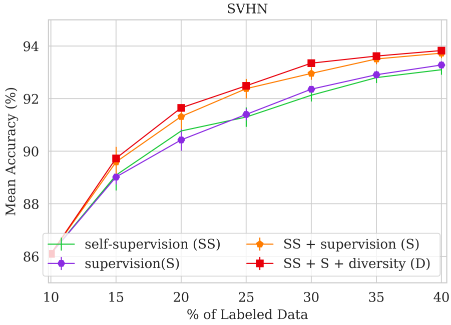

Figure 5: Comparison of PAL's components for scoring unlabeled samples : using only self-supervision ($\lambda_1 = 0, \lambda_2 = 0$), only supervision ($\lambda_1 > 0, \lambda_2 = 0, S_S$ removed), self-supervision & supervision ($\lambda_1 > 0, \lambda_2 = 0$), and self-supervision with supervision & sub-query-based diversity ($\lambda_1 > 0, \lambda_2 > 0$)

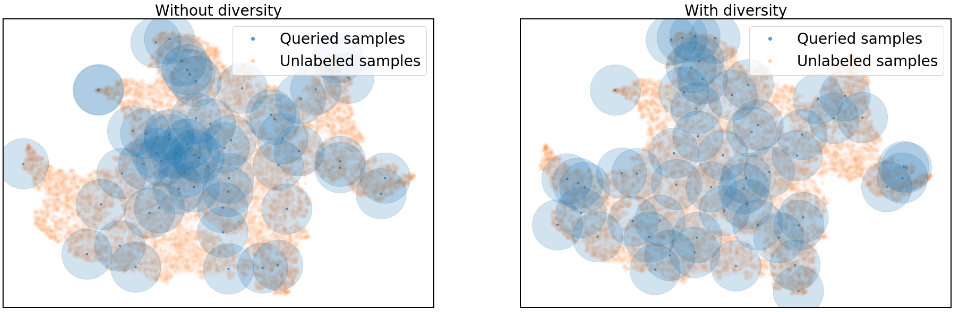

Figure 6: Comparing coverage (blue circles) of the unlabeled data (orange points) from CIFAR-10 using VGG-16 and PAL-based query points (blue points) selected (a) without diversity, shows crowding and (b) with diversity, shows better coverage

4.4 Ablation study

We examined the effect of the different components of the proposed score in Equation 4 used to formulate the query by performing an ablation study. We compared performance by dropping the diversity score S_D ($\lambda_2 = 0, \lambda_1 > 0$), dropping both the diversity score S_D and the supervision score S_C ($\lambda_1 = 0, \lambda_2 = 0$), dropping the self-supervision score and the diversity score ($\lambda_1 > 0, \lambda_2 = 0, S_S$ removed), and the original scenario with both diversity and supervision included ($\lambda_1 > 0, \lambda_2 > 0$). We observed that using uncertainty estimates from both the pretext and classification tasks gave a much better performance. Adding the diversity score resulted in a further improvement in the performance. These results are shown in Figure 5, suggesting that all the three components are important.

We visually show that dividing the query into sub-queries indeed increases the diversity of the query. Figure 6 shows two t-sne embedding plots [31] for CIFAR-10 dataset using a VGG-16 network trained on 10% of data. The unlabeled samples are shown in orange color and the selected query points in blue. A blue circle of arbitrary but same radius for both figures was included for each query point centered at its location to visualize its sphere of coverage. Without diversity, there are gaps in coverage in some areas whereas crowding in some other areas. With diversity, the query points are more spread out providing better coverage of the unlabeled points.

5 Conclusions

We proposed a new pool-based active learning method that is robust to partial mislabeling of the training data, while also giving competitive results for the correctly labeled data. It uses a separate sample scoring network that is dedicated to uncertainty estimation and is trained using self-supervised learning to reduce the dependence on potentially mislabeled data. This work also presents evidence that over-reliance on only one measure of uncertainty may not be judicious. Towards this end, it takes a multi-task approach by introducing a classification head in the scoring network. It also strikes a needed balance between novelty and diversity by ensuring the latter between sub-queries. In general, the separate goals of pursuing novelty and diversity for active learning need more careful integration in future studies.

References

[1] Dana Angluin. Queries and concept learning. In *Machine Learning*, volume 2, 1988.

[2] William H. Beluch, Tim Genewein, Andreas Nürnberger, and Jan M. Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[3] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation, 2017.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020.

[5] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[6] L. Atlas D. Cohn and R. Ladner. Improving generalization with active learning. In *Machine Learning*, volume 15, 1994.

[7] Sanjoy Dasgupta, Daniel J Hsu, and Claire Monteleoni. A general agnostic active learning algorithm. In J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 353–360. Curran Associates, Inc., 2008. URL `http://papers.nips.cc/paper/3325-a-general-agnostic-active-learning-algorithm.pdf`.

[8] Melanie Ducoffe and Frédéric Precioso. Adversarial active learning for deep networks: a margin based approach. *CoRR*, abs/1802.09841, 2018. URL `http://arxiv.org/abs/1802.09841`.

[9] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Pattern Recognition Wkshp*, 2004.

[10] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data, 2017.

[11] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=S1v4N2l0-.

[12] Ran Gilad-bachrach, Amir Navot, and Naftali Tishby. Query by committee made real. In Y. Weiss, B. Schölkopf, and J. C. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 443–450. MIT Press, 2006. URL http://papers.nips.cc/paper/2916-query-by-committee-made-real.pdf.

[13] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 9758–9769. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/8183-deep-anomaly-detection-using-geometric-transformations.pdf.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning, 2020.

[16] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 41–50, 2019.

[17] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks, 2016.

[18] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty, 2019.

[19] M. Huijser and J. C. V. Gemert. Active decision boundary annotation with deep generative models. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5296–5305, 2017.

[20] Alex Krizhevsky. Learning multiple layers of features from tiny images, 2009.

[21] G. Larsson, M. Maire, and G. Shakhnarovich. Colorization as a proxy task for visual understanding. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 840–849, 2017.

[22] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 105–114, 2017.

[23] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. URL http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

[24] M. Noroozi and P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.

[25] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

[26] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks, 2019.

[27] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1aIuk-RW.

[28] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009. URL http://axon.cs.byu.edu/~martinez/classes/778/Papers/settles.activelearning.pdf.

[29] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[30] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *J. Mach. Learn. Res.*, 2:45–66, March 2002. ISSN 1532-4435. doi: 10.1162/153244302760185243. URL https://doi.org/10.1162/153244302760185243.

[31] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[32] Donggeun Yoo and In So Kweon. Learning loss for active learning. 2019.

[33] X. Zhang, J. Zou, K. He, and J. Sun. Accelerating very deep convolutional networks for classification and detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(10):1943–1955, 2016.

[34] Jia-Jie Zhu and José Bento. Generative adversarial active learning. *CoRR*, abs/1702.07956, 2017. URL http://arxiv.org/abs/1702.07956.