Human-in-the-Loop ML Project

Luca Pantea
14651769

Abstract

In this research project, we explore the application of active learning in the context of image data, particularly focusing on scenarios characterized by limited sample sizes and constrained opportunities for querying data points. This study seeks to address the practical challenges faced in real-world settings where data availability is often restricted. Central to our investigation is the examination of uncertainty-based methods within deep learning frameworks. While many active learning acquisition functions depend on model uncertainty, deep learning methods are not typically designed to represent this uncertainty explicitly. Our research aims to empirically test and analyze how these methods perform under the specified conditions, contributing to the understanding of active learning's efficacy in deep learning with image data in constrained environments. This work aims to provide insights and potential directions for the development of more robust and efficient active learning strategies in the context of deep learning, especially when dealing with limited data resources. The code is available on GitHub[1].

1 Introduction

Deep Learning has been widely regarded as the primary driving force behind the most recent significant advancements in Artificial Intelligence. Specifically, deep architectures have been extensively studied in computer vision and have brought significant performance improvements in an array of tasks, such as image classification [1, 20, 26], segmentation [22], detection [34] and retrieval [3]. Much of the success of these models is attributed to the vast amount of data used during training, which ranges from a few thousand to tens of millions of samples [28] in most recent years. While the process of obtaining the data is cost-efficient and fast, acquiring and annotating the data with labels is a tedious labour-intensive task which requires expert human involvement. This comes at a significant cost, both in terms of monetary resources and time.

Active Learning (AL) techniques aim at alleviating the amount of manual annotation required to train machine learning models by iteratively selecting the most informative unlabeled samples and retraining the original model, thereby reducing the demand for annotated data to only a fraction of the original dataset. Originally introduced in Cohn et al. [5], and known as "experimental design" in statistics, has been applied to a wide array of areas, such as NLP [35], medical diagnosis [2], manufacturing and computer vision tasks, such as image/video categorization, text/Web classification, and image/video retrieval [27, 29].

Although active learning has proven effective in multiple domains, various studies [4] have highlighted the scaling issues related to datasets with high-dimensional datasets. When a model is trained using active learning, it has to learn from a significantly lower quantity of samples, and more importantly, learn to represent uncertainty over unseen data [7]. To address these challenges, researchers have explored Bayesian deep learning [7] for more effective handling of high-dimensional mini-batch samples in active learning contexts. Additionally, methods like data augmentation using

[1] `https://github.com/lucapantea/active-learning`

generative networks and the assignment of pseudo-labels to high-confidence samples are being used to improve the quality of the labelled training set [30]. Improvements in batch sample query strategies, considering both information quantity and diversity, are also under active research, aiming to make the DeepAL model adaptable across various fields [16, 36].

However, these advancements encounter challenges when applied to real-world datasets, which often contain inherent data noise due to factors such as varying quality of data collection methods, environmental variables affecting data capture, or the use of automated data gathering tools. This data noise can significantly impact the performance of AL methods relying on deep algorithms, yet most existing literature tends to focus on ideal, noise-free environments. This study addresses these challenges by concentrating on a more constrained AL setting, characterized by two key aspects: i) the use of a smaller initial labelled pool and a reduced validation set size, testing the models' adaptability and efficiency in data-limited situations; ii) a limit on the number of labelled data points queried in each AL round; and iii) the presence of noise patterns in the data. Our work is structured around two research questions:

- **[RQ1]**: The efficiency of Querying Strategies: Between Random, Maximum Entropy, and BALD [21, 11] AL strategies, which is most effective in handling limited sample sizes and noise-influenced image datasets in AL?
- **[RQ2]**: Adaptability and Performance of Deep Learning Models: Does LeNet or ResNet18 demonstrate superior adaptability and effectiveness in AL environments with noisy image data?

The aim is to empirically evaluate how DAL methods fare under these conditions, enhancing our understanding of AL's applicability and efficacy in deep learning for image data in settings that are both constrained and realistic.

2 Background & Related Work

Pool-based Active Learning operates on the principle of iteratively retraining a predictive model by querying the most informative data points from an unlabeled pool set. Initiated with a pre-trained model $p(y|x, \theta_{t=0})$ on a labeled dataset D_0, the model employs an *acquisition function* $\mathcal{A}(x; \theta_t)$ to identify and label new data points x^*, augmenting the training set D_{t+1} and updating the model parameters θ_{t+1}. The process is repeated until the model reaches a satisfactory performance, the pool set is empty, or no further queries can be made to the oracle. More formally,

$$\theta_{t+1} = \text{Train}(D_{t+1}), \quad D_{t+1} = D_t \cup \{(x^*, y^*)\}, \quad x^* = \arg\max_{x \in \mathcal{X}_P^t} \mathcal{A}(x; \theta_t), \tag{1}$$

where y^* is the label provided by the oracle for x^*, and $\mathcal{X}_P^{t+1} = \mathcal{X}_P^t \setminus \{x^*\}$ represents the updated pool set after removing the newly labelled point. For the remainder of the paper, we will primarily consider the pool-based active learning scenario, meaning that when referring to "AL" we mean the pool-based formulation.

Image Classification using Deep Learning Image classification in visual recognition focuses on understanding and assigning a specific label to an image in its entirety. Deep learning approaches, particularly Convolutional Neural Networks (CNNs) [18], have become the standard for this task due to their ability to automatically learn hierarchical feature representations from large datasets, leading to accuracy in various applications.

Active Learning for Image Classification Active Learning (AL) is widely used in image classification, including in areas like medical and scene classification [14, 8]. There are three main types of AL strategies: informativeness [32], representativeness [24], and hybrid methods [12]. A detailed review of these methods is in [23]. Earlier, AL for image data mostly used kernel-based methods. Joshi et al. [13] used SVMs with different kernels for "margin-based uncertainty". Li & Guo [19] applied Gaussian processes with RBF kernels, using simpler features like SIFT. Zhu et al. [37] worked with Gaussian random field models, using RBF kernels on raw images. This early work connected AL with semi-supervised learning [15]. Studies like Gupta et al. [9] and Younesian et al. [33] have made progress in deep AL, especially with noisy labels and using different types of oracles for label quality. The proposed methods outperform the reported standard benchmarks in datasets like MNIST,

CIFAR10, and SVHN, combining model uncertainty, diversity, and information gain for sample selection.

3 Methodology

3.1 Dataset Overview

We perform all of our experiments on three standard benchmark datasets: MNIST [6], CIFAR10 [17] and FashionMNIST [31]. MNIST, a collection of handwritten digits in 28x28 grayscale format. CIFAR10, consisting of 60,000 32x32 colour images in 10 classes, offers a more challenging classification task due to its colour and complexity. FashionMNIST, a dataset comprising of Zalando's article images, serves as a direct but more advanced replacement for MNIST in terms of complexity and variety. More details of the datasets can be found in Table 1.

Table 1: Dataset Statistics for MNIST, CIFAR-10, and FashionMNIST

Dataset	Resolution (Height × Width)	Channels	Description	Dataset Size
MNIST	28x28	1 (Grayscale)	Handwritten Digits	Training: 60,000 Testing: 10,000
CIFAR-10	32x32	3 (RGB)	10 Object Classes	Training: 50,000 Testing: 10,000
FashionMNIST	28x28	1 (Grayscale)	Fashion Items	Training: 60,000 Testing: 10,000

All models are trained on the datasets with an **initial pool of 100 samples** (balanced and random), and a **validation set of 300 samples** on which the the weights are optimized. This is a key aspect of the study, particularly aiding in answering the second research question, since we are examining the effectiveness and robustness of the DL algorithms under the AL framework using limited sample sizes, thus forming a realistic assumption of the validation set size, in comparison to the standard size of 5-7K used commonly on MNIST/CIFAR10 benchmarks. Furthermore, we employ the standard test size of 10K points for evaluating performance, and the rest of the points are part of the unlabelled pool set.

3.2 Models Overview

LeNet LeNet [18], a convolutional neural network designed for digit and image recognition, consists of alternating layers of convolution and subsampling, followed by 3 fully connected layers, using a ReLU activation function throughout. Its less complex structure is effective for low-resolution image processing with limited computational resources and sparse data, making it a suitable candidate for the AL scenario.

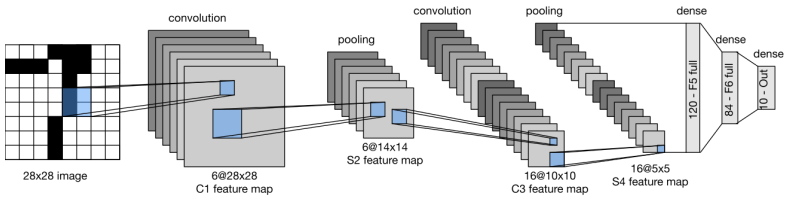


Figure 1: A visualisation of the LeNet-5 network architecture. Image source

ResNet18 ResNet18 [10] is part of the Residual Network family, recognized for its effectiveness in processing high-resolution images. It features 18 layers, including convolutional, batch normalization,

ReLU activation, and pooling layers, leading to a final fully connected layer for classification. The defining aspect of ResNet18 is its use of skip connections, which allow the network to skip over certain layers, effectively addressing the vanishing gradient problem and facilitating the training of deeper models. The model is suitable for this experimental setup, as it offers a more complex alternative to the LeNet architecture, containing residual connections which ensure a more stable and robust training.

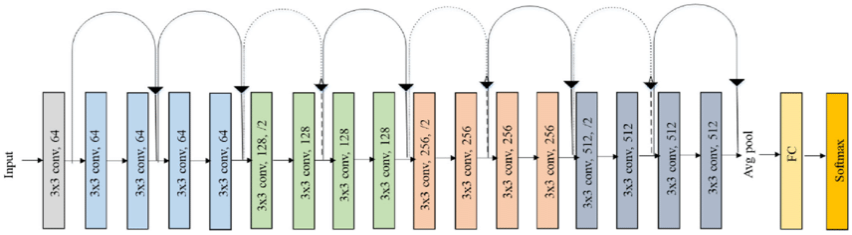


Figure 2: A visualisation of the ResNet18 architecture [10]

3.3 Training Objective

For every active learning round performed, the image classification models are re-trained via a Cross-Entropy objective function, which is defined as follows:

$$\mathcal{L}(\theta) = -\sum_{i=1}^{N} y_i \log(p(y_i|\mathbf{x}_i;\theta)) + (1-y_i)\log(1-p(y_i|\mathbf{x}_i;\theta)), \tag{2}$$

where N is the number of samples, y_i is the true label, $p(y_i|\mathbf{x}_i;\theta)$ is the predicted probability of the label y_i given the input $\mathbf{x}_i$, and θ represents the model parameters. For optimization, we use Stochastic Gradient Descent (SGD) with a weight decay. The parameter update rule in SGD is given by:

$$\theta_{t+1} = \theta_t - \eta\left(\nabla_\theta \mathcal{L}(\theta_t) + \lambda\theta_t\right), \tag{3}$$

where θ_t and θ_{t+1} are the parameters at iterations t and $t+1$, respectively, η is the learning rate, and $\nabla_\theta \mathcal{L}(\theta_t)$ is the gradient of the loss function with respect to the parameters at iteration t.

3.4 Evaluation Metrics

For assessing our model's performance, we primarily utilize two metrics: *validation accuracy* and *test error*. Validation accuracy is calculated at the end of each active learning round. This metric is crucial as it provides immediate feedback on the model's performance with the newly acquired and labelled data, helping us to analyse the effectiveness of the active learning cycle. It also guides the selection of samples in subsequent learning rounds, ensuring a data-efficient learning process. The test error, computed on a separate test set, serves as an indicator of the model's generalization ability.

3.5 Comparison of Acquisition Functions

As specified in Section 2, for a given pre-trained model $p(y|x,\theta_{t=0})$ on a labelled dataset D_0, the model employs an *acquisition function* $\mathcal{A}(x;\theta_t)$ which decides which subsets of points to query next: $x^* = \arg\max_{x\in\mathcal{X}_P^t} \mathcal{A}(x;\theta_t)$. We will explore three acquisition functions in our experiments:

Random Acquisition: This function selects samples randomly from the pool of unlabeled data $\mathcal{X}_P$. It serves as a baseline, assuming no information about the model's uncertainties or the data distribution.

Maximum Entropy: This function selects samples that maximize the entropy of the predictive distribution, $\mathcal{H}[y|x,D_0]$, indicating the model's uncertainty about the prediction [25]. Higher entropy implies less certainty, and thus these points are deemed valuable for querying. Formally:

$$\mathcal{H}[y|x, D_0] := -\sum_c p(y = c|x, D_0) \log p(y = c|x, D_0). \tag{4}$$

Bayesian Active Learning by Disagreement: his function measures the mutual information between the model parameters and the predicted output, as described by Houlsby et al. [11]. It focuses on samples where the current model parameters lead to the greatest disagreement in the predictive distribution, implying a high potential for model improvement upon acquiring the true label. Formally:

$$\mathbb{I}\left[y, \theta \mid x, D_t\right] = \mathbb{H}\left[y \mid x, D_t\right] - \mathbb{E}_{p(\theta|D_0)}[\mathbb{H}[y \mid x, \theta]] \tag{5}$$

Points that yield high values for this acquisition function are those where the model is most uncertain, yet certain model parameters lead to predictions with high confidence that conflict with one another.

3.6 Handling Data Noise in Active Learning

To simulate real-world conditions in our active learning framework, we integrate noise augmentation into our data preprocessing. Specifically, we introduce Additive Gaussian Noise and Additive Salt-and-Pepper Noise to our image datasets. Gaussian noise is applied using the `AddGaussianNoise` class, injecting random noise based on a normal distribution with a specified mean $\mu_{\text{noise}} = 0$ and unit standard deviation σ_{noise}, thereby simulating environmental interference. Salt-and-Pepper noise, added via the `AddSaltAndPepperNoise` class, randomly alters pixel values to the extremes, resembling sensor noise or digital interference. In our experiments, the chances of the noise being added to a datapoint are 10%.

4 Experimental Setup

We evaluate the adaptability of LeNet and ResNet18 models to noisy environments by comparing their active learning performance across three benchmark datasets (detailed in Section 3.1). Each dataset undergoes different noise configurations: no noise, Gaussian noise with a mean of 0 and a unit variance, and Salt-and-Pepper noise affecting a specified percentage of the total pixels. We assess the models using three distinct acquisition strategies (Random Sampling, Maximum Entropy, and BALD), as outlined in Section 3.5.

For each experimental condition, we perform runs with three different random seeds to ensure statistical reliability. The results are then averaged, and the standard error is reported.The active learning loop is executed for 100 rounds, with the model querying 10 new samples from the unlabeled pool in each round. Post-query, models are trained for 100 epochs utilizing an SGD optimizer set with a learning rate of $1e^{-3}$, weight decay of $1e^{-4}$, and a batch size of 64. Data standardization is applied to each dataset utilizing its specific mean and standard deviation. The experiments were conducted using four NVIDIA A100s. We ensure reproducibility by fixing and logging the random seed at the beginning of our experiments and providing detailed versioning of our software environment.

5 Results and Discussion

The findings from our experiments are organized to address the two main research questions stated in the introduction. We have split this section into two parts for clarity.

5.1 Efficiency of Querying Strategies in Constrained Active Learning Settings

In our study, we perform an analysis of the efficiency of different Querying Strategies in Active Learning (AL), focusing on scenarios with limited sample sizes. We aim to identify which strategy—Random, Maximum Entropy, or Bayesian Active Learning by Disagreement (BALD)—performs best under these constraints. While datasets like MNIST, FashionMNIST, and CIFAR10 are popular in vision model research due to their clarity and simplicity, they fall short in replicating the complexity, imperfections and scarcity of real-world annotated data. Recognizing

this limitation, our research emphasizes the need for testing AL strategies in more challenging and realistic conditions, where data is sparse.

To address this, we simulate environments with constrained data availability, mirroring practical challenges faced in real-world applications, as outlined in Section 3.1. We compare the performance of our Active Learning (AL) methods with traditional models that don't use AL but have the same amount of labelled data. For example, if an AL model starts with 100 labelled images and selects 10 new images in each of 100 rounds, it ends up using 1,100 data points in total. We also test an "ideal" scenario where each model begins with 1,000 labelled samples and adds 1,000 more in each of 10 rounds, making 11,000 data points available in total. This approach helps us understand how well AL methods work compared to standard models under different data availability conditions. The data for the Experiments is showcased in Table 3.

Technique	**Test error**
Supervised Learning with limited data:	
LeNet	65.49%
ResNet-18	7.69%
*Active Learning under **ideal** data scenario:*	
LeNet	
Random	4.44%
Max. Entropy	2.39%
BALD	3.37%
ResNet-18	
Random	2.6%
Max. Entropy	1.47%
BALD	2.36%
*Active learning under **constrained** data scenario:*	
LeNet	
Random	71.21%
Max. Entropy	80.94%
BALD	79.23%
ResNet-18	
Random	6.84%
Max. Entropy	8.63%
BALD	6.83%

Figure 3: **Comparison of test error rates for various models and Active Learning strategies (MNIST)**. Supervised Learning is assessed with a training set of 1.1K and a validation set of 300. The "ideal" scenario uses a 1K initial pool with 1K points queried each round of 10, alongside a 5K validation set. The "constrained" scenario starts with a 100-point pool, adds 10 points for each round of 100, and employs a 300-point validation set. Test set: 10K images.

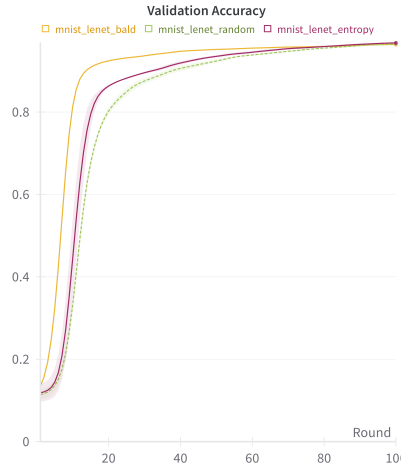
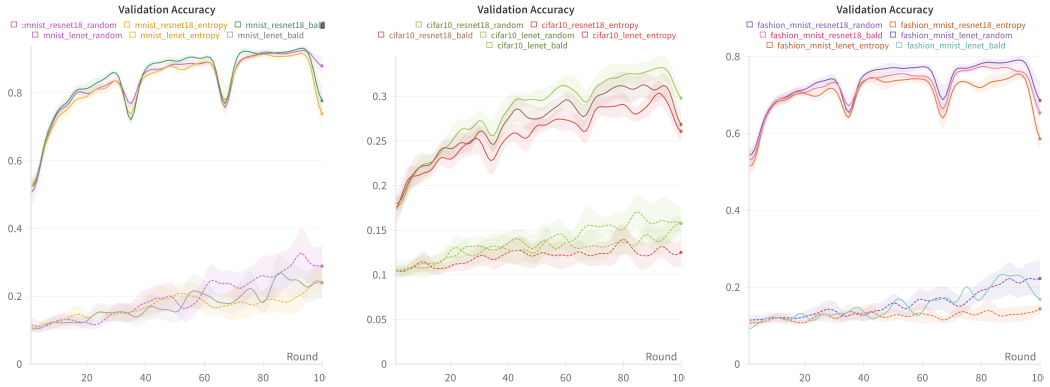


Figure 4: **MNIST validation accuracy in an ideal environment.** The initial labelled pool is set to 1K samples (1.7% of the training dataset), the validation size is 5K, and in each round, 100 samples are queried.

In ideal conditions with more extensive labelled data, Maximum Entropy stands out, particularly for LeNet, decreasing the test error to 2.39%, while for ResNet-18, it achieves an even lower error of 1.47%. This superiority suggests that Maximum Entropy effectively utilizes larger datasets to select informative samples for training. The validation accuracy across rounds can also be observed in Figure 4.

Under constrained data conditions, however, the Random strategy shows an unexpected robustness for ResNet-18, with the lowest error rate at 6.84%. This may imply that in a scenario with sparse data, a less complex strategy like Random can be more beneficial, potentially due to a broader exploration of the sample space. Comparatively, LeNet's performance is noticeably worse in the limited data context across all strategies, with test errors exceeding 70%. This indicates a limitation in LeNet's capacity to generalize from small datasets, an area where ResNet-18 demonstrates more resilience, maintaining lower error rates.

One consistent observation for ResNet-18 is the occurrence of dips in validation accuracy across all three datasets, as seen in Figure 5. These periodic declines may reflect the model's adjustment



(a) Validation accuracy across AL rounds for **MNIST**.

(b) Validation accuracy across AL rounds for **CIFAR10**.

(c) Validation accuracy across AL for **Fashion-MNIST**.

Figure 5: **Validation Accuracy on the three datasets as a function of AL rounds**, for LeNet and ResNet-18 models, using three acquisition functions (*Random*, *Maximum Entropy* and *BALD*). Over the rounds, the models acquire 1000 new data points (100 rounds x 10 queried samples), the validation size is 300 and the accuracy is averaged over 3 random seeds. The standard error spread is highlighted.

phases as it incorporates new, possibly noisy or complex, data points that momentarily challenge its predictive capabilities. A key takeaway from our study is the significant disparity in AL strategy performance between data-rich and data-scarce scenarios. This disparity underscores the need for AL models and strategies that retain efficacy even when data is scarce.

5.2 Active Learning Model Performance under Noisy Data

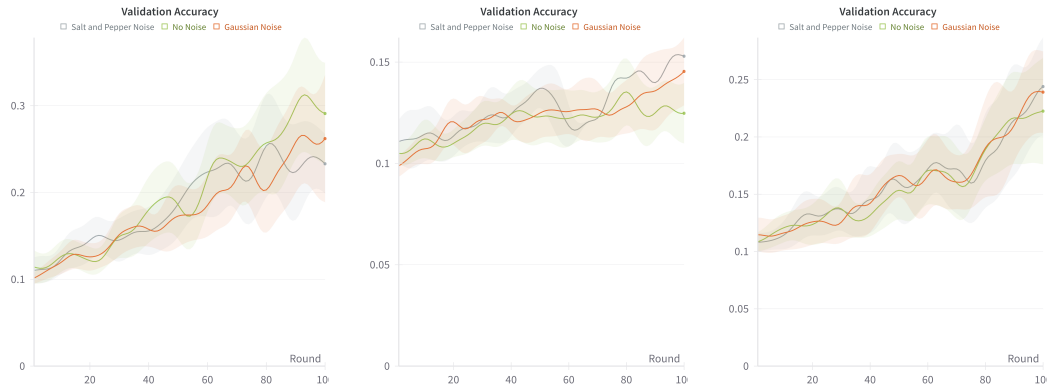
Adding data noise rather than label noise to AL scenarios is crucial for testing the robustness of AL strategies under conditions commonly encountered in real-world datasets. Unlike the bulk of the research that concentrates on label noise, this approach explores the impact of noisy inputs on model training and performance, an aspect that is often faced when applying AL in practical settings.

In our second research question, we investigate the effectiveness of AL strategies in the face of data noise by introducing two types of noise to the three image datasets under study. Gaussian Noise, which simulates random environmental interference with a mean of zero and a standard deviation of one, and Salt-and-Pepper Noise, which simulates random sensor or digital artefacts by altering pixel values to their extremes. Each noise type has a 10% probability of affecting a given data point. The performance of the AL algorithms is given in Table 2 and in Figure 6.

Table 2: Comparison of test accuracy rates across models and Active Learning strategies on MNIST, CIFAR10, and Fashion MNIST datasets with and without Gaussian noise.

Noise	Model	MNIST			CIFAR10			Fashion MNIST		
		Random	Max Entropy	BALD	Random	Max Entropy	BALD	Random	Max Entropy	BALD
None	LeNet	28.79%	19.06%	20.77%	13.92%	13.10%	15.02%	22.99%	19.01%	16.22%
	ResNet-18	91.47%	91.32%	93.16%	33.04%	30.42%	32.23%	77.99%	75.57%	76.19%
Gaussian	LeNet	31.62%	19.38%	42.79%	15.76%	15.68%	17.44%	25.98%	14.63%	17.11%
	ResNet-18	93.44%	94.45%	95.00%	35.21%	34.16%	34.04%	78.47%	74.72%	77.73%
Salt and Pepper	LeNet	18.03%	30.70%	35.74%	14.86%	17.41%	19.04%	20.74%	11.35%	22.20%
	ResNet-18	92.72%	93.17%	94.61%	34.38%	33.31%	38.52%	78.69%	76.79%	78.55%

For the no-noise condition, ResNet-18 showed significantly higher error rates on MNIST, with over 91% error across all strategies, while LeNet maintained much lower error rates (around 19% to 28%). However, in the presence of Gaussian and Salt-and-Pepper noise, ResNet-18's performance was more consistent across all datasets and noise types, maintaining error rates of around 33% to 95%, suggesting a higher resilience to noise. In contrast, LeNet's performance varied more significantly with the type of noise introduced. For instance, under Gaussian noise on MNIST, the error rate



(a) Validation accuracy across AL rounds for noisy configurations of **MNIST**.

(b) Validation accuracy across AL rounds for noisy configurations of **CIFAR10**.

(c) Validation accuracy across AL rounds for noisy configurations of **Fashion-MNIST**.

Figure 6: **Validation Accuracy on the three datasets as a function of AL rounds** for LeNet under three different noise configurations in the data (Absent, Unit Gaussian and Salt and Pepper), using the *BALD* acquisition functions.

for LeNet with the BALD strategy spiked to 42.79%, indicating a susceptibility to this type of noise. However, LeNet performed better under the Salt-and-Pepper noise condition, especially in the CIFAR10 and Fashion MNIST datasets, with error rates ranging from 11.35% to 22.20%.

For MNIST, the presence of noise generally results in lower validation accuracy compared to the no-noise scenario. However, in later rounds, the accuracy for models trained with noise shows an improving trajectory, suggesting that the AL process is adapting to the noise. Interestingly, in the case of salt-and-pepper noise, the validation accuracy surpasses that of the Gaussian noise after a certain number of rounds, which could indicate that the model finds it easier to adapt to the more binary nature of Salt-and-Pepper noise over the continuous variations introduced by Gaussian noise, but further experimentation would need to be carried out, as the spreads of the standard error intersect.

CIFAR10 presents a more challenging scenario for both models, with generally lower accuracy levels. ResNet-18's accuracy is consistent under noise conditions, with a less pronounced accuracy dip than LeNet. This could suggest that ResNet-18's deeper architecture captures more complex representations that are less perturbed by noise, thus maintaining a steadier performance.

These results suggest that while ResNet-18 demonstrates a consistent performance across different noise conditions, LeNet's adaptability varies depending on the type of noise, indicating that the choice of model in AL environments should be contingent on the specific nature of data noise expected in the application.

6 Conclusions, Limitation, & Future Work

Our study investigated active learning strategies in limited sample size environments and noisy data conditions, focusing on the performance of querying strategies and the adaptability of two deep learning models - LeNet and ResNet-18. The results suggest varying effectiveness of the Random, Maximum Entropy, and BALD strategies depending on data availability. In environments with limited data, the Random strategy showed an unexpected performance for ResNet-18. Conversely, in data-rich conditions, Maximum Entropy was more effective. The study also indicated that the adaptability of deep learning models to noise varies, with ResNet-18 showing more consistent performance across different noise conditions compared to LeNet.

Limitations The study's reliance on simulated environments and specific types of noise may not fully reflect the complexities of real-world datasets. The focus on uncertainty-based methods within deep learning frameworks and the limited selection of deep learning models, namely LeNet and ResNet-18, could restrict the generalizability of the findings. Furthermore, the scope of noise types examined was limited, potentially overlooking other real-world data disturbances and noisy labels.

Future Work Incorporating real-world datasets with a broader range of noise types to better understand the effectiveness of active learning strategies in practical applications. It would be interesting to test a wider variety of deep learning models to ensure the generalizability of the results. Additionally, investigating the mechanisms behind different models' adaptability to noise could lead to improvements in active learning methodologies

References

[1] Carlos Affonso, André Luis Debiaso Rossi, Fábio Henrique Antunes Vieira, and André Carlos Ponce de Leon Ferreira de Carvalho. Deep learning for biological image classification. *Expert Systems with Applications*, 85:114–122, 2017.

[2] Angona Biswas, MD Abdullah Al Nasim, Md Shahin Ali, Ismail Hossain, Dr. Md Azim Ullah, and Sajedul Talukder. Active learning on medical image, 2023.

[3] Wei Chen, Yu Liu, Weiping Wang, Erwin M. Bakker, Theodoros Georgiou, Paul W. Fieguth, Li Liu, and Michael S. Lew. Deep image retrieval: A survey. *CoRR*, abs/2101.11282, 2021.

[4] Gui Citovsky, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar. Batch active learning at scale. *CoRR*, abs/2107.14263, 2021.

[5] David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. Active learning with statistical models. *CoRR*, cs.AI/9603104, 1996.

[6] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[7] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. *CoRR*, abs/1703.02910, 2017.

[8] E. Gavves, T. Mensink, T. Tommasi, C. G. M. Snoek, and T. Tuytelaars. Active transfer learning with zero-shot priors: Reusing past datasets for future tasks. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2731–2739, 2015.

[9] Gaurav Gupta, Anit Kumar Sahu, and Wan-Yi Lin. Learning in confusion: Batch active learning with noisy oracle. *CoRR*, abs/1909.12473, 2019.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[11] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning, 2011.

[12] Sheng-Jun Huang, Rong Jin, and Zhi-Hua Zhou. Active learning by querying informative and representative examples. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(10):1936–1949, 2014.

[13] Ajay J. Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2372–2379, 2009.

[14] Ashish Kapoor, Kristen Grauman, Raquel Urtasun, and Trevor Darrell. Active learning with gaussian processes for object categorization. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007.

[15] Diederik P. Kingma, Danilo Jimenez Rezende, Shakir Mohamed, and Max Welling. Semi-supervised learning with deep generative models. *CoRR*, abs/1406.5298, 2014.

[16] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *CoRR*, abs/1906.08158, 2019.

[17] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[18] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[19] Xin Li and Yuhong Guo. Adaptive active learning for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013.

[20] Òscar Lorente, Ian Riera, and Aditya Rana. Image classification with classic and deep learning techniques. *CoRR*, abs/2105.04895, 2021.

[21] David J. C. MacKay. Information-Based Objective Functions for Active Data Selection. *Neural Computation*, 4(4):590–604, 07 1992.

[22] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *CoRR*, abs/2001.05566, 2020.

[23] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *CoRR*, abs/2009.00236, 2020.

[24] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach, 2018.

[25] Claude Elwood Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 1948.

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[27] Simon Tong. Active learning: theory and applications. 2001.

[28] Pablo Villalobos and Anson Ho. Trends in training dataset sizes, 2022. Accessed: 2024-01-09.

[29] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *CoRR*, abs/1701.03551, 2017.

[30] Wenzhe Wang, Ruiwei Feng, Jintai Chen, Yifei Lu, Tingting Chen, Hongyun Yu, Danny Z. Chen, and Jian Wu. Nodule-plus r-cnn and deep self-paced active learning for 3d instance segmentation of pulmonary nodules. *IEEE Access*, 7:128796–128805, 2019.

[31] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

[32] Donggeun Yoo and In So Kweon. Learning loss for active learning. *CoRR*, abs/1905.03677, 2019.

[33] Taraneh Younesian, Dick Epema, and Lydia Y. Chen. Active learning for noisy data streams using weak and strong labelers, 2020.

[34] Syed Sahil Abbas Zaidi, Mohammad Samar Ansari, Asra Aslam, Nadia Kanwal, Mamoona Naveed Asghar, and Brian Lee. A survey of modern deep learning based object detection models. *CoRR*, abs/2104.11892, 2021.

[35] Zhisong Zhang, Emma Strubell, and Eduard Hovy. A survey of active learning for natural language processing, 2023.

[36] Fedor Zhdanov. Diverse mini-batch active learning. *CoRR*, abs/1901.05954, 2019.

[37] Xiaojin Zhu, John Lafferty, and Zoubin Ghahramani. Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. 08 2003.