A Survey of Active Learning Sampling Strategies for Image Classification

Dongxuan He
Department of Engineering
Purdue University
he711@purdue.edu

Pengyuan Guo
College of Enginerring
Purdue University
guo464@purdue.edu

Teresa Huang
Department of Computer Science
Purdue University
huan1606@purdue.edu

Wei Wu
School of Const. Mgmt. Tech.
Purdue University
wu2424@purdue.edu

Yi Ding
Department of Computer Science
Purdue University
ding432@purdue.edu

Abstract

Active learning (AL) seeks to reduce annotation cost by iteratively querying labels for the most informative unlabeled examples. In this project, we conduct a controlled empirical comparison of several representative AL acquisition strategies on Fashion-MNIST. We consider a pool-based AL loop, retraining the model from scratch each round to reflect the evolving labeled set. The classifier is a lightweight multilayer perceptron, followed by a linear softmax head optimized with cross-entropy and Adam. Empirically, BALD attains the highest test accuracy across labeling budgets, while BADGE, CSAL, and standard uncertainty heuristics (least confidence, margin, entropy) achieve comparable but slightly lower performance. Core-Set exhibits weaker accuracy than the uncertainty-driven methods and, in our experiments, can be comparable to or below uniform random sampling. We also observe a consistent accuracy–efficiency trade-off: methods that require additional stochastic forward passes or optimization in embedding space incur higher computational cost, whereas simple uncertainty scoring remains relatively inexpensive.

1 Introduction

Active learning (AL) is a class of methods that seek high predictive performance while querying as few labels as possible (Ren, et al. 2021 [5]). Instead of labeling an entire dataset in advance, the learner starts with a small labeled set, trains a model, and then repeatedly selects additional unlabeled examples to be annotated by an oracle (e.g., a human annotator). By focusing the annotation effort on samples that promise the greatest information gain, AL can significantly lower the human labeling burden while still maintaining strong performance.

In a typical pool-based active learning protocol, the process begins with a small initial set of labeled instances and a large pool of unlabeled data. A model is trained on the labeled set, and then an acquisition function (query strategy) evaluates the unlabeled pool to identify the most informative sample(s) to label next (Wu, 2019 [8]). Often these are points about which the current model is most uncertain. The selected data points (the query instances) are sent to an oracle (e.g. a human annotator) for labeling and then added to the training set. The model is retrained with this augmented labeled set, and the cycle repeats. Through each iteration, the model improves as new informative labels are incorporated, until a stopping criterion is met (such as a labeling budget or desired accuracy) (Ren,

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

et al. 2021 [5]). This iterative refine-and-query loop lies at the core of AL, enabling the model to progressively improve with minimal label expenditure.

In this project, Fashion-MNIST serves as a testbed to evaluate how various active learning approaches perform in terms of label efficiency and accuracy improvement. This approach provides insight into the relative strengths of AL methods before deploying them on domains where labeling truly is costly. A wide range of query selection strategies have been developed for active learning, we will be focusing on 5 main methods in this study.

2 Related Works

2.1 Bayesian Active Learning [2] by Gal et al., 2017:

BALD (Bayesian Active Learning by Disagreement) is a principled Bayesian framework for active learning that aims to select unlabeled samples whose labels would maximally reduce the model's parameter uncertainty. Formally, BALD selects points that maximize the mutual information between the predicted label Y and the model parameters W conditioned on the input x,

$$I(Y, W \mid x, \mathcal{D}),$$

which measures how informative the label of x would be for updating the model.

Deep neural networks make exact Bayesian inference intractable, so Gal et al. (2017) propose using *MC Dropout* as an efficient approximation to the posterior. By keeping dropout active at test time and performing multiple stochastic forward passes, the model produces a set of predictive distributions that approximate samples from the posterior predictive distribution.

BALD quantifies uncertainty by measuring the amount of *disagreement* between these stochastic predictions. Samples for which different dropout realizations predict conflicting outcomes exhibit high epistemic uncertainty and therefore yield large expected information gain when labeled. This distinguishes BALD from classical uncertainty metrics (e.g., entropy) that conflate model uncertainty with data noise (aleatoric uncertainty). Empirically, BALD has been shown to be highly effective in low-label regimes and is widely regarded as one of the strongest Bayesian acquisition strategies for deep active learning.

2.2 BADGE: Batch Active Learning by Diverse Gradient Embeddings [1] by Ash et al., 2020

Classical uncertainty sampling often relies on entropy over the model's predictive distribution, which assumes access to well-calibrated posterior probabilities. However, this assumption does not always hold in practice. Entropy-based querying is not directly applicable to non-probabilistic learners, such as standard SVMs (Zhu, et al. 2010 [10]). Representation-based scores such as EGL, LSA, and DSA may yield strong gains on image classification but degrade performance on text classification (Hu, et al. 2021 [4]). Recent research has accordingly turned toward hybrid approaches that seek a trade-off between querying highly uncertain samples and diverse samples in each batch.

Batch Active Learning by Diverse Gradient Embeddings (BADGE) was proposed to address the above limitations by unifying uncertainty and diversity in a single, simple method (Ash, et al., 2020 [1]). The core idea in BADGE is to use the model's gradient information as a dual-purpose representation of each unlabeled example. One reflects both how uncertain the model is about that example, the other represents how different the example is from others. In BADGE, for each candidate data point, we first computes a gradient embedding (the gradient of the model's loss with respect to the parameters of the last layer). Intuitively, this gradient vector points in the direction the model's weights would move if the example's (hypothetical) label were revealed. Then, we will apply a k-means++ clustering in the space of these gradient embeddings to select a batch of points that are maximally spread out (diverse) in that space. Overall, each chosen point has a large gradient norm (signaling high informativeness) while the set of chosen points are as different from each other as possible (ensuring coverage of different regions or uncertainties).

Specific goal for this project: By re-implementing BADGE and evaluating it on representative tasks, we seek to verify the original results and assess the method's behavior in a controlled setting. Our goal is to confirm BADGE's reported advantages that it can achieve comparable or superior accuracy to standard active learning strategies.

2.3 Active Learning for Convolutional Neural Networks: A Core-Set Approach [6] by Sener et al., 2018

Active learning serves as the primary paradigm to address this labeling bottleneck. The fundamental goal is to develop smart strategies for selecting the most informative images from a very large unlabelled collection. However, traditional active learning heuristics face significant hurdles when applied to modern deep learning.

Ineffectiveness in Batch Settings: Empirical studies suggest that many established active learning heuristics are ineffective when applied to CNNs in a batch setting. **The "Batch" Constraint:** In classical settings, algorithms typically select a single data point at each iteration, but this is infeasible for CNNs because a single point is unlikely to have a statistically significant impact on accuracy due to local optimization methods, and each iteration requires full training until convergence, making it intractable to query labels one by one. **Correlation Issues:** Consequently, CNNs require acquiring labels for large subsets (batches) at each iteration. The authors argue that the main factor behind the ineffectiveness of standard heuristics is the correlation caused by this batch sampling; heuristics often select redundant, correlated points rather than a diverse set.

A New Perspective: Core-Set Selection Inspired by these limitations, the authors redefine the active learning problem as core-set selection. The objective shifts from finding "uncertain" points to finding a subset of points such that a model learned over this selected subset is competitive with a model learned over the remaining data points.

Differentiation from Prior Work This work distinguishes itself from existing literature in three key ways: it explicitly defines active learning as a core-set selection problem; it considers both fully supervised and weakly supervised learning cases; and it addresses the core-set selection problem rigorously and directly for CNNs without extra assumptions. While the problem of core-set selection exists in literature (e.g., for SVMs or k-Means), those methods consider fully labeled datasets and attempt to compress them. The authors note they are unaware of such methods previously existing for CNNs.

2.4 Uncertainty-based Sampling, A survey of deep active learning [5] by Ren, P., Xiao, et al., 2021:

Active Learning (AL) strategies aim to maximize model performance while minimizing annotation costs by iteratively querying the most informative samples. Uncertainty-based sampling remains the most prevalent baseline due to its computational efficiency and intuitive rationale [5]. These methods operate on the principle that samples near the decision boundary—where the model exhibits low predictive confidence—provide the highest gradient information for model updates.

Three metrics define this category:

Least Confident (LC): Queries samples where the model's probability for the most likely class is lowest. While computationally inexpensive, LC ignores the distribution of the remaining classes.

Margin Sampling: Queries samples with the smallest difference between the top-two class probabilities. This method directly targets decision boundary ambiguity, making it particularly effective for multi-class classification.

Entropy Sampling: Uses Shannon entropy to measure global uncertainty across the entire predictive distribution. While theoretically rigorous, entropy-based methods are often susceptible to querying outliers or noisy samples in high-dimensional spaces [5].

Recent comparative surveys indicate that while sophisticated methods like BADGE incorporate diversity to prevent sampling redundancy, simple margin-based strategies often remain competitive baselines in standard classification tasks.

2.5 Consistency-based Semi-supervised Active Learning (CSAL) [3]

Active Learning (AL) improves data efficiency by selectively querying labels for informative unlabeled samples. However, standard AL pipelines often overlook the rich information contained in the unlabeled pool during training. CSAL addresses this limitation by incorporating a semi-supervised objective based on self-consistency, encouraging the model to produce stable predictions under data

augmentation. In addition, CSAL employs the self-consistency score as its selection criterion: samples with low consistency are treated as highly uncertain and are prioritized for labeling in subsequent iterations.

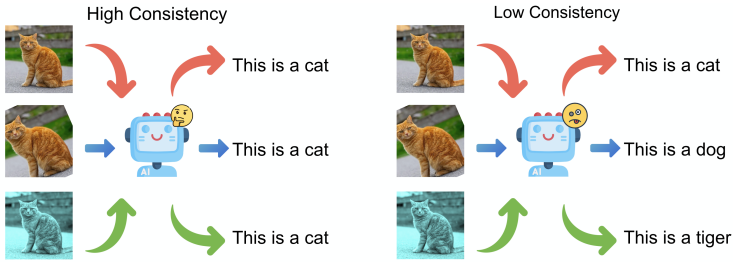

Figure 1: Illustration of model self-consistency.

Self-Consistency. Self-consistency quantifies a model's confidence on a given sample by evaluating whether it produces consistent predictions under small perturbations [7]. A reliable model should remain stable under mild augmentations, as shown on the left, where the model consistently predicts the image as a cat. Conversely, samples with low self-consistency behave as hard examples: they induce high uncertainty and thus benefit most from being annotated, making them ideal candidates for acquisition in AL.

3 Problem Definition: Active Learning on Fashion-MNIST

Active Learning Framework We consider the problem of multi-class image classification on the Fashion-MNIST dataset, defined over an input space $\mathcal{X} = \mathbb{R}^{28\times 28}$ and a label space $\mathcal{Y} = \{0, \ldots, 9\}$. We assume access to a large pool of unlabeled data $\mathcal{U}$ and a small initial set of labeled data $\mathcal{L}_0$, where $|\mathcal{L}_0| = 500$.

The goal of the active learning algorithm is to iteratively select a batch of b unlabeled points to query from an oracle (human annotator) to maximize the performance of a classifier f_θ trained on the updated labeled set. In our experimental setup, the query budget is fixed at $b = 100$ samples per round for a total of 100 rounds.

4 Method

4.1 BADGE

BADGE Embedding In this subsection we describe how these gradient embeddings are defined and computed using the MLP architecture introduced above. Recall that for an input image x, the network produces a 256-dimensional penultimate representation $\mathbf{h}(x) \in \mathbb{R}^{256}$. For a labeled example (x, y), the network is trained using the sparse categorical cross-entropy loss

$$\ell(x, y; \theta) = -\log p_y(x).$$

BADGE extends this notion of loss to **unlabeled** points by introducing a **hallucinated label**. At a given active learning round, with current parameters θ, we first predict the most likely class $\hat{y}(x) = \arg\max_{k\in\{1,\ldots,K\}} p_k(x)$, and then treat $\hat{y}(x)$ as a pseudo-label. The corresponding loss for an unlabeled point is

$$\ell(x, \hat{y}(x); \theta) = -\log p_{\hat{y}(x)}(x).$$

The key idea in BADGE is to represent each unlabeled example x by the gradient of this hallucinated loss with respect to the parameters of the final layer. For the softmax classifier described above, the gradient of $\ell(x, \hat{y}(x); \theta)$ with respect to the row $W_i^{(2)}$ associated with class i has the closed form

$$\frac{\partial \ell(x, \hat{y}(x); \theta)}{\partial W_i^{(2)}} = \big(p_i(x) - \mathbf{1}[\hat{y}(x) = i]\big)\, \mathbf{h}(x), \qquad i = 1, \ldots, K,$$

where $\mathbf{1}[\hat{y}(x) = i]$ is the indicator that the predicted class equals i. Each class-specific gradient is therefore a scaled copy of the penultimate embedding $\mathbf{h}(x)$, with the scaling determined by the softmax probabilities and the predicted label.

To obtain a single vector representation, BADGE stacks these class-specific gradients into a single gradient embedding. In practice, this is implemented by computing the gradient of the hallucinated cross-entropy loss with respect to $W^{(2)}$ and then flattening the resulting $K \times 256$matrix into a vector.

Intuitively, the **norm** $\|g(x)\|$ captures how much the output layer would change if the (pseudo-)label for x were used for training; points with large gradient norms are therefore informative in the sense of inducing large parameter updates. The **direction** of $g(x)$ encodes both the feature embedding $\mathbf{h}(x)$ and the pattern of class probabilities $\mathbf{p}(x)$, so two points that would push the model in very different directions have nearly orthogonal gradient embeddings.

k-MEANS++ Seeding After computing gradient embeddings for the entire unlabeled pool in a given round, we run the k-MEANS++ seeding algorithm in this embedding space to select a batch of query points, as suggested by the authors. The seeding procedure iteratively chooses new centers with probability proportional to the squared distance from the closest already-selected center, favoring points both far and with high-magnitude gradient embeddings.

4.2 CSAL

Each iteration of CSAL comprises two stages, training and selection:

Training. Conventional AL methods rely exclusively on labeled data, leaving the information in unlabeled data unused. CSAL instead incorporates a semi-supervised learning (SSL) objective at every AL cycle. The target model M_t is optimized with a combined loss $L_l + L_u$, where L_l is the supervised loss (e.g., cross-entropy) and L_u is an unsupervised consistency loss:

$$\mathcal{L}_u(x, M) = D\Big(P(\hat{Y} = l \mid x, M),\ P(\hat{Y} = l \mid \tilde{x}, M)\Big),$$

where $\tilde{x}$ is an augmented view of x, and D is a distance metric (we use KL divergence [9]). This consistency objective enables the model to exploit unlabeled samples by encouraging stable predictions across augmentations.

Selection. A core challenge in AL is identifying which unlabeled samples are most beneficial for subsequent training. Prior work typically selects samples with high uncertainty, but the reliability of such estimates is not explicitly enforced. CSAL addresses this gap by using the self-consistency score—the same metric optimized during training—to measure a model's uncertainty. Samples with the lowest consistency scores are selected, yielding a more principled and robust acquisition strategy.

Data Splitting. At iteration t, let $\mathcal{D}_l^{t-1}$ and $\mathcal{D}_u^{t-1}$ denote the labeled and unlabeled datasets. After incorporating previously selected samples, the updated labeled set becomes $\mathcal{D}_l^t = \mathcal{D}_l^{t-1} \cup \mathcal{D}_s^{t-1}$. We then draw a batch of b unlabeled samples, $\mathcal{D}_b^t$, from $\mathcal{D}_u^{t-1}$ for consistency training. During selection, we compute self-consistency scores on the remaining unlabeled set $\mathcal{D}_u^t = \mathcal{D}_u^{t-1} \setminus \mathcal{D}_b^t$ and acquire the lowest-scoring samples for the next iteration.

4.3 Uncertainty

We implemented an iterative AL framework on the Fashion-MNIST dataset using a "cold start" protocol. The initial labeled pool $\mathcal{L}_0$ consisted of 100 randomly selected images, with the remaining data forming the unlabeled pool $\mathcal{U}$.

Uncertainty Metrics. We formulated three uncertainty acquisition functions. Let $\hat{y}$ denote the class with the highest probability for input x, and $P(y|x;\theta)$ be the predicted probability under model parameters θ.

Least Confident: Selection focuses on the worst-case probability:

$$\phi_{LC}(x) = 1 - P(\hat{y}|x;\theta)$$

Margin Sampling: Selection targets the ambiguity between the most probable class $\hat{y}_1$ and the second most probable class $\hat{y}_2$:

$$\phi_{Margin}(x) = P(\hat{y}_1|x;\theta) - P(\hat{y}_2|x;\theta)$$

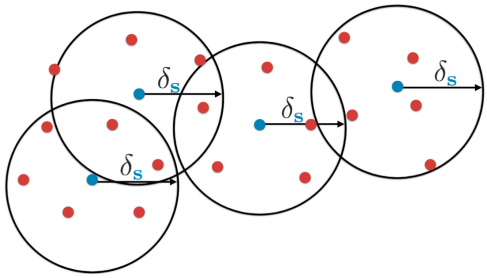

Figure 2: Visualization of Theorem 1. Consider the set of selected points $\int$ and the points in the remainder of the dataset $[n] \setminus \int$, our results show that if $\int$ is the δ_s-cover of the dataset.

We minimize this margin (or maximize its inverse) to identify samples near the decision boundary.

Entropy: Selection maximizes the information content of the predictive distribution:

$$\phi_{Entropy}(x) = -\sum_{y} P(y|x;\theta) \log P(y|x;\theta)$$

4.4 CORE-SETS FOR CNNS

Core-Set Formulation Following the Core-Set approach, we re-frame the active learning problem as a geometric covering problem rather than an uncertainty estimation problem. The objective is to select a set of points that minimizes the covering radius δ of the entire dataset in the learned feature space.

Let $h(x)$ denote the feature representation of an input x extracted from the penultimate layer of the neural network. We define the problem as finding a subset of points $S \subset \mathcal{U}$ of size b such that the maximum distance from any point in the dataset to its nearest labeled neighbor is minimized. This is formally equivalent to the k-Center problem (Min-Max Facility Location):

$$\min_{S:|S|\leq b} \max_{x_i \in \mathcal{U}} \min_{x_j \in \mathcal{L}_t \cup S} ||h(x_i) - h(x_j)||_2 \tag{1}$$

Since this problem is NP-Hard, we employ the *k-Center Greedy* approximation algorithm. At each step, the algorithm selects the point $u \in \mathcal{U}$ that maximizes the distance to the current labeled set $\mathcal{L}$:

$$u = \arg\max_{x_i \in \mathcal{U}} \min_{x_j \in \mathcal{L}} \Delta(h(x_i), h(x_j)) \tag{2}$$

where Δ represents the Euclidean distance in the feature embedding space.

4.5 BALD

MC Dropout for Bayesian Approximation. Bayesian Active Learning by Disagreement (BALD) [2] is a principled acquisition strategy that measures how much an unlabeled example *would* reduce the model's epistemic uncertainty if its label were revealed. Since exact Bayesian neural networks are intractable for modern architectures, BALD uses **Monte Carlo dropout** to approximate sampling from the posterior over model weights. During inference, dropout layers are left active, producing a stochastic model $f^{(t)}$ at each forward pass. For an unlabeled input x, the model generates T predictive distributions

$$p^{(t)}(y \mid x), \qquad t = 1, \ldots, T,$$

which approximate samples from the posterior predictive distribution.

Acquisition Function: Mutual Information. BALD selects examples that maximize the mutual information between predictions and model parameters:

$$I(Y, W \mid x, \mathcal{D}),$$

where Y is the predicted label, W represents the model weights, and $\mathcal{D}$ is the labeled dataset. Intuitively, BALD scores how much the label of x would help reduce uncertainty about the model itself.

The mutual information can be computed in closed form as

$$\mathrm{BALD}(x) = H(\,\overline{p}(y \mid x)\,) - \frac{1}{T}\sum_{t=1}^{T} H\Big(p^{(t)}(y \mid x)\Big)\,,$$

where $\overline{p}(y \mid x) = \frac{1}{T}\sum_t p^{(t)}(y \mid x)$ is the averaged predictive distribution, and $H(p)$ denotes entropy. The first term measures total predictive uncertainty, while the second term reflects expected aleatoric uncertainty. The difference isolates **epistemic uncertainty**, which is precisely the component that active learning aims to reduce. Examples that yield high disagreement across the stochastic forward passes receive high BALD scores and are prioritized for annotation.

5 Experiment

5.1 Experimental Setting

To evaluate this formulation, we utilize the following specific parameters:

Dataset: Fashion-MNIST (FMNIST), consisting of 28x28 grayscale images of clothing items.

Model Architecture: For all experiments we use a small multilayer perceptron (MLP). The network operates on grayscale images $x \in \mathbb{R}^{28\times 28}$. We first flatten the image into a vector

$$\mathbf{v} = \mathrm{vec}(x) \in \mathbb{R}^{784}.$$

The hidden (penultimate) representation $\mathbf{h} \in \mathbb{R}^{256}$ is computed by an affine transformation followed by a ReLU nonlinearity:

$$\mathbf{h} = \phi\Big(W^{(1)}\mathbf{v} + \mathbf{b}^{(1)}\Big)\,,$$

where $W^{(1)} \in \mathbb{R}^{256\times 784}$, $\mathbf{b}^{(1)} \in \mathbb{R}^{256}$, and $\phi(a) = \max(a, 0)$ is applied elementwise. This 256-dimensional vector $\mathbf{h}$ serves as the shared feature embedding. The output layer is a linear classifier without bias followed by a softmax. Given a labeled dataset $\{(x_n, y_n)\}_{n=1}^{N}$ with $y_n \in \{1, \ldots, K\}$, we train the network by minimizing the sparse categorical cross-entropy loss

$$\mathcal{L}(\theta) = -\frac{1}{N}\sum_{n=1}^{N} \log p_{y_n}(x_n; \theta),$$

where $p_{y_n}(x_n; \theta)$ denotes the predicted probability of the ground-truth class for example n. Optimization is performed using the Adam optimizer with a learning rate of 3×10^{-4}, and we monitor classification accuracy on a held-out test set during training.

Random sampling baseline. To contextualize the benefit of informed acquisition functions, we include *random sampling* as a simple reference baseline. At each active learning round, the baseline selects a batch of unlabeled instances uniformly at random from the current unlabeled pool (without replacement).

Training Protocol: In each active learning round, the model is re-initialized and trained until convergence (defined as reaching 99% training accuracy or hitting 1000 epochs) to ensure the feature space reflects the current labeled distribution.

5.2 Main Results

Accuracy. As shown in Figure 3, BALD consistently outperforms other active learning strategies, achieving the highest test accuracy as more samples are acquired. Margin Sampling, BADGE, CSAL, Entropy Sampling,and Least Confident form a strong second tier, showing similar performance with relatively fast early-stage gains. In contrast, Coreset performs even worse than random sampling, especially in the early iterations, suggesting less effective sample selection for Fashion-MNIST.

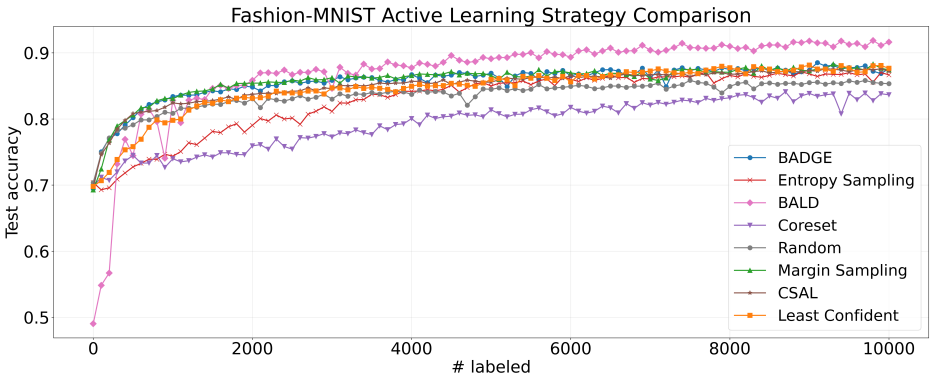

Figure 3: Training curves of different methods in 10000 iterations.

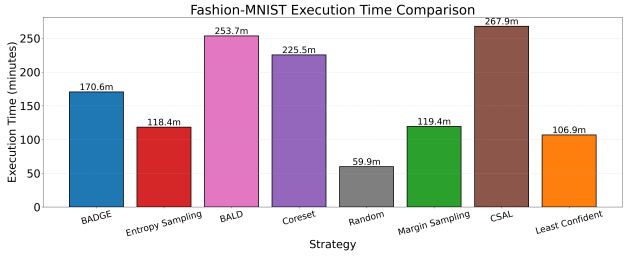

Figure 4: Comparison of training time.

Efficiency. BALD, CSAL, and Coreset introduce additional training objectives, which incur substantial computational overhead. In contrast, sampling-based methods—whether using entropy, confidence, or margin—require far less computation. As a result, they add only minimal training cost compared with the random-select baseline and thus provide a highly efficient alternative.

5.3 More Analysis

5.3.1 Why Core-Set Failed: The Critical Role of Feature Geometry

The Discrepancy: Theory vs. Our Results A significant divergence exists between the theoretical promises of Core-Set selection and our experimental findings. In the original paper, the Core-Set method (red line) significantly outperforms both Random and Uncertainty baselines. However, in our reproduction, Core-Set performance drops below that of Random sampling. The key difference driving this discrepancy is the model architecture: the paper utilized VGG-16, a deep CNN, whereas our experiment utilized a simple MLP.

The Root Cause: Geometric Dependence The core idea of the algorithm is to select points that minimize the "covering radius" (δ) based on Euclidean distance in the feature space. This approach fundamentally assumes that the distance between points correlates with semantic difference. The paper explicitly notes that "better feature spaces result in accurate geometries," implying that the method's success is contingent on the quality of the underlying representation.

The Failure Mode: Picking Outliers over Diversity This reliance on geometry explains the failure mode. CNNs generate a robust feature space where the "farthest" points are semantically distinct and informative. Conversely, MLPs generate a noisy feature space on image data. In this suboptimal space, the "farthest" points selected by the Greedy k-Center algorithm are often statistical outliers or "garbage" data rather than informative examples. By aggressively targeting these outliers, the Core-Set method with an MLP actively harms performance compared to Random sampling, which avoids these edge cases.

5.3.2 Uncertainty

We evaluated the efficacy of the three uncertainty-based acquisition functions—Least Confident, Margin Sampling, and Entropy—on the Fashion-MNIST dataset. The experiment was conducted

over 100 active learning rounds with a query budget of 100 samples per round, resulting in a total labeled pool of 10,000 samples by the final iteration.

Comparative Performance Dynamics As illustrated in Figure 5, the learning trajectories reveal distinct behavioral characteristics among the three metrics:

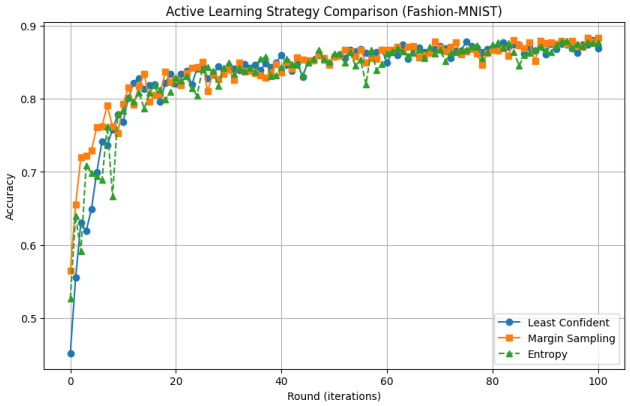

Figure 5: Schematic illustration of the Uncertainty-based Active Learning workflow.

The iterative process begins with a cold-start labeled pool ($\mathcal{L}_0$). A Convolutional Neural Network (CNN) is trained to output class probabilities for the unlabeled pool. Three distinct acquisition functions—Least Confident, Margin Sampling, and Entropy—are employed to quantify predictive uncertainty. The top-K instances maximizing these scores are queried for annotation and added to the next labeled set ($\mathcal{L}_{t+1}$).

Robustness of Margin Sampling: Margin Sampling (Orange) : demonstrated the most consistent performance gain, particularly in the critical early learning phase (Rounds 0–20). By explicitly targeting the probability gap between the top-two predicted classes, this strategy effectively isolates samples residing on the decision boundaries of visually similar categories (e.g., Shirt vs. T-shirt). This focus allows the model to resolve inter-class ambiguity more efficiently than metrics that consider only the single highest probability.

Volatility of Entropy: Entropy Sampling (Green) : exhibited high volatility, characterized by sharp performance oscillations (e.g., the significant drop near Round 8). While Entropy captures global distributional uncertainty, this sensitivity acts as a double-edged sword: it is prone to querying "outliers" or noisy samples that maximize information content mathematically but degrade the decision boundary empirically.

Baseline Performance of Least Confident: generally followed the trend of the other metrics but often lagged slightly in stability during intermediate rounds. Since it discards information regarding the second-best and remaining classes, it lacks the discriminative power needed to fine-tune the model as effectively as Margin Sampling in complex multi-class scenarios.

Convergence and Saturation. Despite the differences in early-stage efficiency, all three strategies achieved asymptotic convergence, reaching a test accuracy of approximately 88% by Round 100. This saturation suggests that as the labeled pool size approaches 10,000 samples, the information redundancy in the remaining unlabeled pool increases, diminishing the marginal utility of specific selection strategies. However, Margin Sampling's superior stability makes it the most reliable baseline for scenarios where annotation budgets are strictly limited to the early, steep region of the learning curve.

5.3.3 BALD

BALD exhibits two distinctive behaviors in our experiments: it is significantly slower than other methods, yet it eventually achieves the best performance. Both effects arise from the use of Monte

Carlo Dropout. Since BALD requires T stochastic forward passes for every unlabeled example (we use $T = 20$), its acquisition step is roughly $20\times$ more expensive than single-pass uncertainty methods, explaining its high computational cost.

Despite this overhead, BALD consistently outperforms all other strategies in later rounds. By estimating epistemic uncertainty through model disagreement, BALD prioritizes samples that truly reduce uncertainty in the model parameters, while naturally avoiding aleatoric noise and outliers that entropy-based methods often select. As the model becomes more stable in later iterations, dropout-induced disagreement becomes reliable, enabling BALD to acquire highly informative samples and achieve the highest accuracy.

Interestingly, BALD starts with noticeably lower accuracy (around 50%). This occurs because, in the early stages, the model is poorly calibrated and its dropout-ensemble behaves almost randomly, making disagreement estimates unreliable. As training progresses and the representation improves, BALD transitions from unstable early behavior to consistently strong performance, ultimately surpassing all baselines.

6 Conclusion

By a systematic comparison, our survey aims to provide practical insights into when and why certain sampling techniques outperform others in real-world deep learning settings. Our experiments validate that Margin Sampling is a robust and cost-effective baseline. It successfully exploits decision boundary ambiguity without the computational overhead of diversity-based embedding methods, although it lacks the robustness to outliers provided by semi-supervised approaches.

7 Contribution

All authors jointly defined the experimental protocol, evaluation metrics, and the shared MLP pipeline for training. Each team member was responsible for implementing, validating, and writing the methodology and discussion of one acquisition strategy.

References

[1] J. T. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

[2] Y. Gal, R. Islam, and Z. Ghahramani. Deep bayesian active learning with image data. In *International conference on machine learning*, pages 1183–1192. PMLR, 2017.

[3] M. Gao, Z. Zhang, G. Yu, S. Ö. Arık, L. S. Davis, and T. Pfister. Consistency-based semi-supervised active learning: Towards minimizing labeling cost. In *European Conference on Computer Vision*, pages 510–526. Springer, 2020.

[4] Q. Hu, Y. Guo, M. Cordy, X. Xie, W. Ma, M. Papadakis, and Y. L. Traon. Towards exploring the limitations of active learning: An empirical study. *IEEE/ACM International Conference on Automated Software Engineering*, 2021.

[5] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B. B. Gupta, X. Chen, and X. Wang. A survey of deep active learning, 2021.

[6] O. Sener and S. Savarese. Active learning for convolutional neural networks: A core-set approach, 2018.

[7] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[8] D. Wu. Pool-based sequential active learning for regression. *IEEE Transactions on Neural Networks and Learning Systems*, 30(5):1348–1359, 2019.

[9] Q. Xie, Z. Dai, E. Hovy, T. Luong, and Q. Le. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 33:6256–6268, 2020.

[10] J. Zhu, H. Wang, B. K. Tsou, and M. Ma. Active learning with sampling by uncertainty and density for data annotations. *IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 6*, 2010.