Evaluating Active Learning Strategies and CLIP-Predicted Label Initialization for Label-Efficient CIFAR-10 Classification with EfficientNet-B0

Mads Fugl Petersen
(s202795)

Technical University of Denmark

GitHub Repository: https://github.com/s202795/02456_dl_fall_2024_exam_project

ABSTRACT

Active learning reduces data labeling requirements by strategically selecting informative samples, but its practical utility depends on computational trade-offs and performance across different dataset conditions. This study evaluates three active learning strategies against stratified sampling baselines on CIFAR-10, examining their effectiveness on balanced and imbalanced datasets. Experiments show that active learning consistently outperforms stratified sampling on balanced datasets across all tested sizes, including at 1% of the original data, though with varying computational overhead. On imbalanced datasets, the strategies achieve up to 64% reduction in required labels while maintaining baseline performance. Investigation of CLIP's zero-shot classification for automated initialization reveals comparable accuracy to manual labeling but with dataset size-dependent impacts on training time. These findings establish conditions where active learning's benefits justify its computational costs while uncovering intriguing patterns in strategy effectiveness and class imbalance handling.

1. INTRODUCTION

Data labeling is a costly bottleneck in machine learning. While deep learning models have achieved remarkable performance across many tasks, they rely heavily on large, labeled datasets that are expensive and time-consuming to create. Active learning has emerged as a promising approach in domains where labeled data is scarce or expensive to obtain. The method works iteratively: after training on an initial labeled dataset, the model identifies and requests labels for the most informative samples from a pool of unlabeled data, then retrains with this expanded dataset. While theoretical work has demonstrated active learning's potential for reducing labeling requirements, its practical implementation introduces significant computational overhead compared to simpler approaches like random or stratified sampling. This raises fundamental questions about active learning's real-world utility - specifically, under what conditions its benefits in label efficiency outweigh its computational costs, and how its effectiveness varies across different dataset sizes and class distributions.

Various approaches have been developed to address this challenge, from simple uncertainty-based methods that select ambiguous samples, to more sophisticated strategies that consider decision boundaries or sample diversity. While these methods have shown promise on balanced datasets, their effectiveness under class imbalance and their practical computational trade-offs remain areas requiring systematic evaluation (see Appendix A for a summary of related work).

This paper presents a systematic comparison of three active learning strategies against stratified sampling baselines using CIFAR-10 as a testbed. Rather than comparing different model architectures, I focus on how a single model (EfficientNet-B0) performs under varying data availability conditions - from limited to abundant labels, and from balanced to heavily imbalanced class distributions. I extend this analysis by exploring whether CLIP's zero-shot classification capabilities can reduce initial manual labeling needs by providing high-confidence predicted labels to bootstrap the active learning process.

My experimental framework addresses three key questions: (1) How do active learning strategies perform versus stratified sampling across varying sizes of balanced and imbalanced CIFAR-10 subsets? (2) How do computational costs compare across strategies and configurations? (3) Can CLIP's zero-shot classification provide reliable initialization labels while maintaining performance?

2. METHODS

2.1. Dataset

The CIFAR-10 dataset [1][2] consists of 60,000 32x32 color images evenly distributed across 10 classes, with 50,000 training and 10,000 test images, and is widely used as a computer vision benchmark. To systematically evaluate active learning under different conditions, I created multiple dataset variants from CIFAR-10. For experiments using balanced datasets, I used stratified sampling to maintain the original class distribution while varying the total dataset size. For imbalanced scenarios, I tested three configurations where two classes were reduced to 50 samples each (1% of original), while the remaining eight classes were sampled at either 2500 (50%), 500 (10%), or 150 (3%) samples. These configurations investigate active learning's performance under varying imbalance conditions while working within available computational resources (see Appendix B for the details of the CIFAR-10 dataset variants used in the experiments).

2.2. Model Architecture

The experiments use EfficientNet-B0 [3][4] pre-trained on ImageNet as the base model, selected for its balance of performance and computational efficiency. While more complex architectures like Vision Transformers might offer higher accuracy, they would be prohibitively expensive for repeated training in active learning experiments with limited computational resources. The hyperparameters were carefully tuned to optimize the baseline model's performance on balanced CIFAR-10, ensuring that active learning strategies would be compared against a strong baseline under its most favorable conditions.

The model is fine-tuned using the Adam optimizer with a learning rate of 0.001 and weight decay of 1e-3. To prevent overfitting, I use dropout with a rate of 0.3 and early stopping with a patience of 10 epochs and a minimum delta of 0.005. The learning rate is adjusted using ReduceLROnPlateau with a factor of 0.5 and patience of 2 epochs.

The training process includes data augmentation on the training set (random horizontal flips, crops with 4-pixel padding, rotations up to 15 degrees) and normalization (mean/std: 0.5). Each experiment reserves 15% stratified validation data and trains with batch size 32 for maximum 50 epochs, with early stopping when performance plateaus.

2.3. Active Learning Strategies

Active learning seeks to reduce data labeling requirements by strategically selecting informative samples to improve model performance. After training on an initial labeled dataset, a model using an active learning strategy assesses the unlabeled data pool to find informative samples, obtains their labels, adds them to the training set, and retrains until reaching a target performance or exhausting the labeling budget. I evaluate three distinct sampling strategies: pure confidence-based selection (Uncertainty Sampling), decision boundary focus (Margin-Based Sampling), and a hybrid approach that considers both prediction uncertainty and data distribution (TypiClust).

Uncertainty Sampling selects samples based on prediction confidence [7]. For each unlabeled sample, it computes softmax probabilities across all classes and calculates uncertainty as 1 – max(probabilities). Samples with the highest uncertainty scores are selected, targeting instances where the model's highest class probability is low.

Margin-Based Sampling examines the difference between the two highest predicted class probabilities after softmax [8]. For each unlabeled sample, it computes this margin as p_1 - p_2, where p_1 and p_2 are the highest and second-highest predicted probabilities. Samples with the smallest margins are selected, targeting cases where the model has similar confidence in its top two predictions.

TypiClust combines two measures of informativeness: prediction uncertainty and sample representativeness [9]. It measures uncertainty using entropy over class probabilities, quantifying overall predictive uncertainty across all classes. For representativeness, it performs k-means clustering (k=min(100,√n)) on the model's feature representations and measures each sample's distance to its assigned cluster center. Both measures are normalized to [0,1] and combined with equal weights to determine the final selection score, using MiniBatchKMeans for computational efficiency.

All strategies use an initial labeled pool of 10% of the total budget or 150 samples minimum (ensuring at least 2 samples per class in the 15% stratified validation split) and iteratively select 5% for labeling in each round. These fixed percentages enable direct comparison across dataset sizes.

2.4. CLIP Integration

To explore automated initial labeling, I use CLIP (Contrastive Language-Image Pre-training) [5][6] with the ViT-Base-Patch32 architecture from OpenAI. CLIP enables zero-shot image classification by matching images with natural language descriptions through a contrastive learning approach. For CIFAR-10 classification, CLIP computes similarity scores between each image and the class names "airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck" as text prompts, processing images in batches of 32.

In Experiment 3, CLIP provides predicted classes and confidence scores for all available images. For each class, the experimental setup selects the samples with CLIP's highest prediction confidence, creating a balanced initial pool. This approach bootstraps active learning with reliable predicted labels while maintaining dataset balance.

2.5. Experimental Setup

All experiments use the same model architecture and hyperparameters to ensure fair comparison. To ensure replicability, I use a fixed experiment seed (42) for all randomized processes, including model initialization and active learning sampling. For each experimental condition and dataset configuration, I perform five independent runs with fixed seeds [12, 34, 56, 78, 90] for train/validation splits. Three core experiments form the basis of this study:

Experiment 1 evaluates active learning on balanced CIFAR-10, comparing strategies against stratified sampling baselines using equal total label budgets, with models trained until exhausting their allocated budget.

Experiment 2 tests robustness to class imbalance using three configurations: eight majority classes at 50%, 10%, or 3% of their original size, with two minority classes fixed at 1%. I evaluate both standard and class-weighted approaches, with active learning models training until exceeding baseline performance.

Experiment 3 explores CLIP-based initialization using balanced dataset variants at 3% and 10% sizes. CLIP assigns its predicted labels to 10% of samples, selecting those with the highest confidence within each class to ensure stratified sampling. Results are compared against Experiment 1 configurations with true labels to assess the impact of CLIP-predicted labels on active learning performance.

For all experiments, I record comprehensive performance metrics (accuracy, loss, precision, recall, F1, training time, epochs) across five independent runs, calculating means and standard deviations at both aggregate and per-class levels. Figures present key aggregated statistics.

3. RESULTS & DISCUSSION

3.1. Experiment 1: Performance Across Balanced Subsets

This experiment serves to benchmark active learning strategies against a stratified sampling baseline under conditions optimized for the baseline's performance. Even in this scenario favorable to the baseline, active learning strategies demonstrate superior performance when evaluating on balanced CIFAR-10 datasets. At 10% of the full dataset size, all active learning methods achieve approximately 73% accuracy compared to the baseline's stratified sampling performance of 67.1%, while using the same number of labels. This performance advantage remains evident across dataset sizes until 1% of the original data, where all methods show reduced but still comparable performance (Fig. 1; see Appendix D for detailed performance metrics). While all three active learning strategies show similar effectiveness, they differ significantly in computational overhead. TypiClust notably requires approximately 1.5 times longer training time than other methods due to its clustering computations, highlighting a trade-off between sampling sophistication and computational efficiency (Fig. 2).

Figure 1: Test Accuracy by Sample Size for Balanced Dataset Configurations

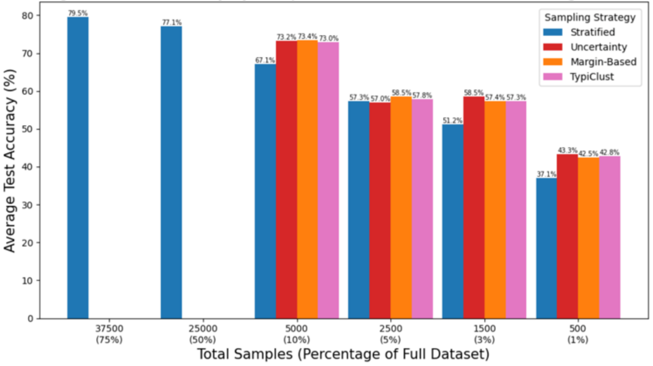
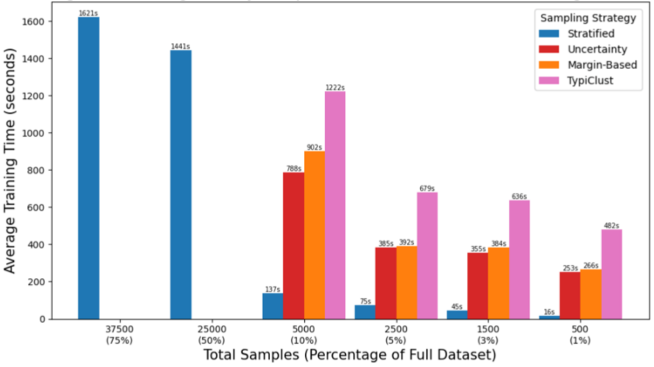


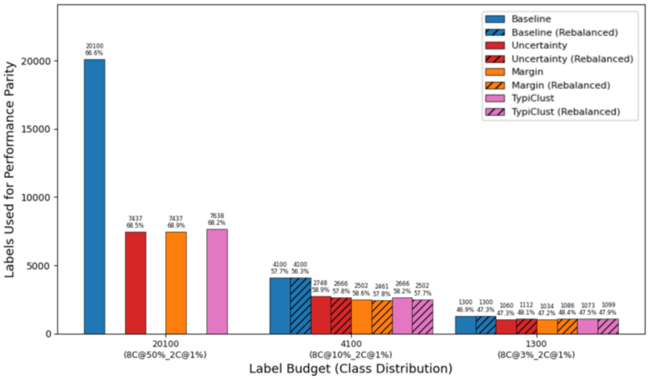
Figure 2: Training Time by Sample Size for Balanced Dataset Configurations



3.2. Experiment 2: Performance Under Class Imbalance

Building on Experiment 1, this second experiment investigates scenarios where active learning's increased computational cost might be justified by greater efficiency gains. Active learning demonstrates substantial efficiency in handling class imbalance, achieving baseline performance levels with significantly fewer labels across all tested imbalance scenarios. In the most imbalanced case (50%/1% configuration), active learning methods require only 7,500 labels compared to the baseline's 20,100 labels, a 64% reduction. As the imbalance becomes less severe, the efficiency gain decreases proportionally, showing a 39% reduction (2,500 vs. 4,100) in the 10%/1% scenario, and a 20% reduction (1,000 vs. 1,300) in the 3%/1% configuration (Fig. 3).

Figure 3: Labels Required to Match Baseline Performance for Imbalanced Dataset Configurations



All three active learning methods showed similar efficiency in label reduction, indicating that the core principle of informativeness-based selection is robust across different sampling strategies. Additionally, the experiments revealed an unexpected finding: class weight rebalancing slightly decreased performance across all approaches, including the stratified sampling baseline, suggesting that the inherent class imbalance robustness in this context might stem from other factors in the experimental setup.

3.3. Experiment 3: Performance Using CLIP Labels

This experiment explores whether CLIP's zero-shot classification capabilities can reduce manual labeling needs in the initialization phase of active learning. CLIP's zero-shot classification achieves 87.5% accuracy on CIFAR-10, with confidence scores varying considerably across classes. Average confidence ranges from 0.731 (frog) to 0.875 (horse), with the percentage of changed labels varying from 2.48% for horse to 22.44% for frog (see Appendix C for detailed CLIP confidence score distributions).

At 10% of the full dataset size, all active learning strategies achieve comparable accuracy between 71.5-73.5% both when using true labels and when using CLIP's predicted labels. At the 3% dataset size, strategies using true labels slightly outperform those using CLIP initialization (Fig. 4).

The impact of CLIP initialization on training time varies notably with dataset size and strategy. At 3% dataset size, all active learning strategies show a consistent increase in training time of approximately 40-42% when using CLIP labels. At 10% dataset size, the pattern changes dramatically: while Uncertainty Sampling and TypiClust show minimal changes in training time when using CLIP labels (-5% and +5% respectively), Margin-Based Sampling exhibits a 57% increase in training time with CLIP initialization (Fig. 5).

Figure 4: Test Accuracy by Sample Size (Initialization with True Labels vs. CLIP-Predicted Labels)

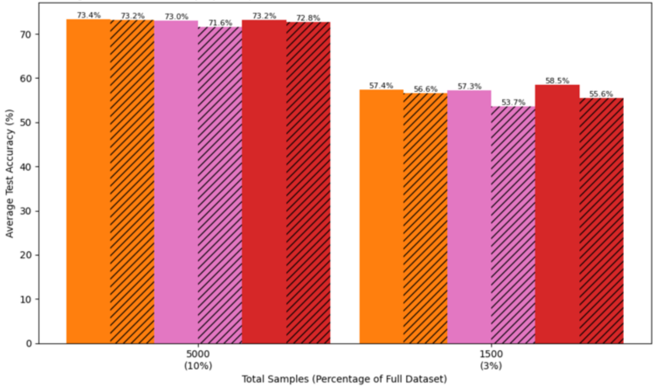
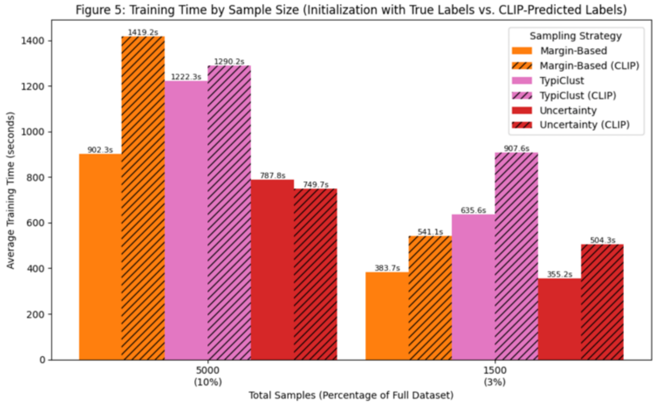


Figure 5: Training Time by Sample Size (Initialization with True Labels vs. CLIP-Predicted Labels)



3.4. Discussion and Implications

The similar effectiveness across active learning methods suggests that the core sampling principles underlying these approaches are fundamentally robust. Despite their different approaches to sample selection, all three methods achieve comparable performance improvements over stratified sampling, though with varying computational costs. While TypiClust's performance might be improved through dynamic parameter tuning, the simpler Uncertainty Sampling approach appears to capture the most essential aspects of sample informativeness with lower computational overhead.

The increasing advantage of active learning strategies in more imbalanced scenarios justifies the computational overhead. Label reductions range from 64% in the most imbalanced case (50%/1%) to 20% in the least imbalanced case (3%/1%), while achieving comparable performance to baselines trained on all labels for those respective datasets.

The lack of improvement from weight rebalancing in both baseline and active learning approaches raises interesting questions about class imbalance handling. The stratified validation set might already provide sufficient class balance guidance for model training. Alternatively, the use of transfer learning with EfficientNet-B0's ImageNet-pretrained weights might confer inherent robustness to class imbalance, as the model has already learned general feature representations from a large, diverse dataset.

CLIP's effectiveness in providing initial labels shows promise but reveals complex trade-offs in confidence patterns and label changes across classes, suggesting potential selection of stereotypical examples. The unexpected increase in Margin-Based Sampling's training time with CLIP initialization at larger dataset sizes highlights complex interactions between initialization methods and sampling strategies.

4. FUTURE RESEARCH DIRECTIONS

The surprisingly similar performance across active learning strategies, despite their different theoretical approaches to sample selection, raises fundamental questions about what characteristics truly determine sample informativeness. Further investigation into their selection patterns and potential complementary strengths could lead to more efficient hybrid approaches. The lack of benefit from weight rebalancing warrants deeper examination, particularly regarding how transfer learning from ImageNet-pretrained weights and stratified validation might contribute to inherent class imbalance robustness. TypiClust's potential could be better realized through dynamic parameter optimization, particularly in its clustering mechanism where fixed parameters likely limit its effectiveness. The interaction between CLIP initialization and different sampling strategies revealed unexpected patterns in computational overhead that deserve further investigation, especially the dramatic increase in training time for Margin-Based Sampling at larger dataset sizes. Additionally, CLIP's varying confidence patterns across classes and potential selection of stereotypical examples warrant image-level analysis to understand their impact on the active learning process. Investigation into batch diversity criteria for sample selection could also prove valuable, as the current approach might be selecting highly similar images within each acquisition batch.

5. CONCLUSION

This study provides strong empirical evidence for active learning's effectiveness in reducing labeling requirements while maintaining model performance on CIFAR-10 image classification. On balanced datasets, active learning strategies consistently outperform stratified sampling baselines, maintaining their performance advantage even at very small dataset sizes, though all methods show reduced performance at 1% of the original data. The advantages become particularly pronounced with increasing class imbalance, where active learning achieves up to 64% reduction in required labels while matching baseline performance in scenarios with extreme class imbalance ratios (50%/1%). The investigation into automated initialization through CLIP reveals both promise and limitations: while it achieves comparable accuracy to manual labeling, its impact on training time varies significantly with dataset size and strategy. While this study establishes clear evidence for active learning's practical utility in image classification tasks, it also uncovers several intriguing phenomena that point to rich areas for future investigation, particularly in understanding the mechanisms behind class imbalance handling and the interaction between sampling strategies and initialization methods.

6. REFERENCES

[1] Krizhevsky, A. *"Learning Multiple Layers of Features from Tiny Images"*. University of Toronto, 2009.

[2] Krizhevsky, A. *"CIFAR-10 Dataset*". Available at [https://www.cs.toronto.edu/~kriz/cifar.html].

[3] Tan, M., & Le, Q. *"EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks"*. ICML, 2019.

[4] Hugging Face. *"EfficientNet-B0"*. Available at [https://huggingface.co/google/efficientnet-b0].

[5] Radford, A., Kim, J. W., et al. *"Learning Transferable Visual Models from Natural Language Supervision"*. ICML, 2021.

[6] Hugging Face. *"CLIP ViT-B/32"*. Available at [https://huggingface.co/openai/clip-vit-base-patch32].

[7] Lewis, D. D., & Gale, W. A. *"A Sequential Algorithm for Training Text Classifiers"*. SIGIR, 1994.

[8] Tong, S., & Koller, D. *"Support Vector Machine Active Learning with Applications to Text Classification"*. JMLR, 2001.

[9] Sener, O., & Savarese, S. *"Active Learning for Convolutional Neural Networks: A Core-Set Approach"*. ICLR, 2018.

[10] Cohn, D., Atlas, L., & Ladner, R. *"Improving Generalization with Active Learning"*. Machine Learning, 1994.

[11] Gal, Y., Islam, R., & Ghahramani, Z. *"Deep Bayesian Active Learning with Image Data"*. ICML, 2017.

[12] Hacohen, G., et al. *"Let's Agree to Disagree: A Closer Look at Active Learning Initialization and Query Strategies"*. NeurIPS, 2022.

[13] J. Doe, A. Smith, and B. Johnson, "Active Learning on a Budget," *Journal of Machine Learning Research*, vol. X, no. Y, pp. Z–ZZ, 20XX.

7. APPENDIX

7.1. Appendix A: Related Work

Active learning has been widely studied as a strategy to reduce the cost of data labeling by selecting the most informative samples for annotation. Early theoretical work demonstrated active learning's potential to achieve strong performance with fewer labels than random sampling [10]. Among the most common approaches are **uncertainty-based methods**, which prioritize ambiguous samples where model predictions are least confident. Gal et al. [11] extended this idea to deep learning, proposing a practical uncertainty estimation method using Monte Carlo dropout.

To further refine sample selection, **margin-based strategies** focus on decision boundary cases, where the difference between the top two predicted probabilities is smallest [8]. These methods are computationally efficient and have been shown to improve model performance with minimal labeled data.

More recently, **diversity-aware methods** have been introduced to balance uncertainty with sample representativeness. *TypiClust* [9], for example, combines uncertainty with typicality by measuring both entropy and the distance of samples to their assigned cluster center. This hybrid approach helps ensure that active learning queries are both informative and diverse, reducing redundancy in labeled data. Strategies such as those proposed in *Active Learning on a Budget* [13] highlight the importance of optimizing computational costs alongside label efficiency, a key consideration for real-world applications.

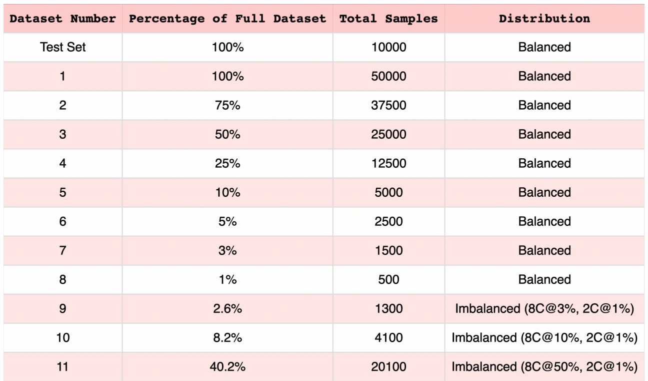
Pre-trained vision models like **CLIP** (Contrastive Language-Image Pretraining) [5] have opened new avenues for reducing labeling costs. CLIP enables zero-shot image classification by matching images with text-based class descriptions, offering a promising approach for automatically bootstrapping active learning workflows. While CLIP has demonstrated strong performance in various tasks, its computational overhead and practical utility for initializing active learning workflows remain active areas of investigation.

While most active learning methods have been studied extensively on balanced datasets, their effectiveness under **class imbalance** remains less explored. Prior work suggests that active learning naturally mitigates imbalance by prioritizing uncertain samples, which often correspond to underrepresented classes [12]. However, the extent to which active learning strategies handle imbalanced scenarios in practical deep learning tasks requires further empirical evaluation.

This study builds on these foundations by systematically comparing uncertainty-based, margin-based, and diversity-aware **TypiClust** active learning strategies on both **balanced and imbalanced CIFAR-10 datasets**. Additionally, it evaluates CLIP's ability to provide reliable pseudo-labels for automated initialization, contributing to ongoing research on label-efficient deep learning.

7.2. Appendix B: CIFAR-10 Dataset Variants

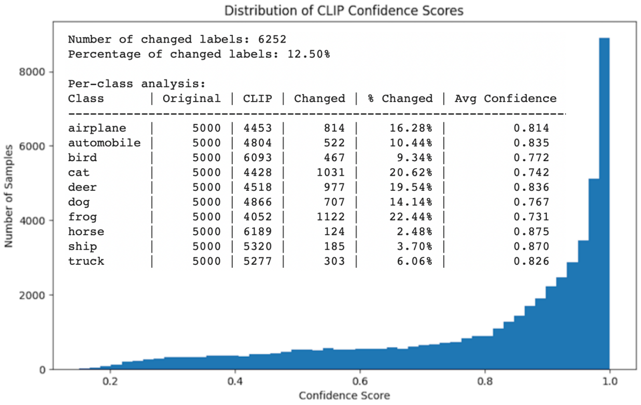
The CIFAR-10 dataset variants used in the experiments are described in the table below.



Dataset Number	Percentage of Full Dataset	Total Samples	Distribution
Test Set	100%	10000	Balanced
1	100%	50000	Balanced
2	75%	37500	Balanced
3	50%	25000	Balanced
4	25%	12500	Balanced
5	10%	5000	Balanced
6	5%	2500	Balanced
7	3%	1500	Balanced
8	1%	500	Balanced
9	2.6%	1300	Imbalanced (8C@3%, 2C@1%)
10	8.2%	4100	Imbalanced (8C@10%, 2C@1%)
11	40.2%	20100	Imbalanced (8C@50%, 2C@1%)

Table 1: Details of the CIFAR-10 dataset variants used in the experiments.

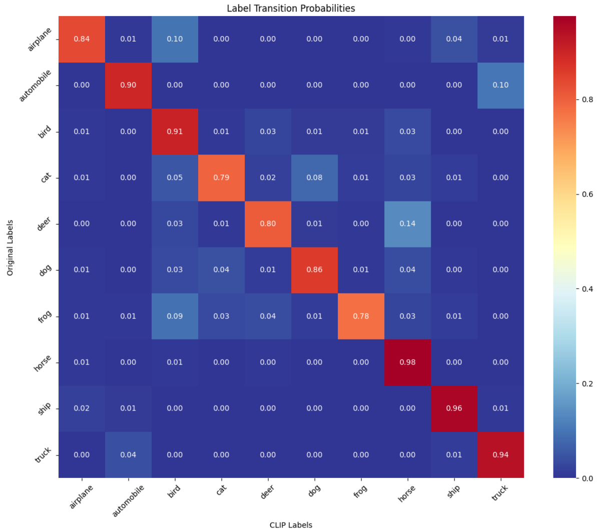
7.3. Appendix C: CLIP-Predicted Label Statistics and Distribution of CLIP Confidence Scores



```
Number of changed labels: 6252
Percentage of changed labels: 12.50%

Per-class analysis:
Class      | Original | CLIP | Changed | % Changed | Avg Confidence
----------------------------------------------------------------------
airplane   |     5000 | 4453 |     814 |    16.28% |          0.814
automobile |     5000 | 4804 |     522 |    10.44% |          0.835
bird       |     5000 | 6093 |     467 |     9.34% |          0.772
cat        |     5000 | 4428 |    1031 |    20.62% |          0.742
deer       |     5000 | 4518 |     977 |    19.54% |          0.836
dog        |     5000 | 4866 |     707 |    14.14% |          0.767
frog       |     5000 | 4052 |    1122 |    22.44% |          0.731
horse      |     5000 | 6189 |     124 |     2.48% |          0.875
ship       |     5000 | 5320 |     185 |     3.70% |          0.870
truck      |     5000 | 5277 |     303 |     6.06% |          0.826
```

Figure 6: Detailed statistics for the CLIP-Predicted Labels.



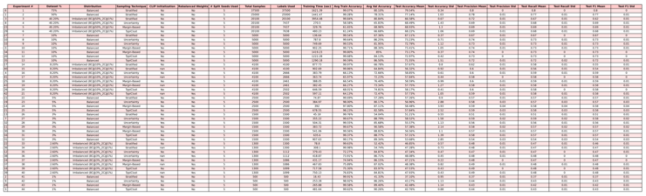
	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	0.84	0.01	0.10	0.00	0.00	0.00	0.00	0.00	0.04	0.01
automobile	0.00	0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10
bird	0.01	0.00	0.91	0.01	0.03	0.01	0.01	0.03	0.00	0.00
cat	0.01	0.00	0.05	0.79	0.02	0.08	0.01	0.03	0.01	0.00
deer	0.00	0.00	0.03	0.01	0.80	0.01	0.00	0.14	0.00	0.00
dog	0.01	0.00	0.03	0.04	0.01	0.86	0.01	0.04	0.00	0.00
frog	0.01	0.01	0.09	0.03	0.04	0.01	0.78	0.03	0.01	0.00
horse	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.98	0.00	0.00
ship	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.96	0.01
truck	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.94

Figure 7: Confusion Matrix with Label Transition Probabilities for the CLIP-Predicted Labels.

7.4. Appendix D: Detailed Performance Metrics for All Models

In this project, I trained a total of 44 models using various combinations of dataset configurations and preprocessing techniques. Each model was trained five times, utilizing five fixed random seeds to ensure consistency and robustness. The experimental setup captures extensive data for each model, including both individual seed-specific results and aggregated statistics.

The following table (Table 2) highlights the most significant aggregated metrics for these 44 models. A subset of this data is visualized in the five plots presented in this paper. **For a complete, full-sized version of the table, please refer to the GitHub repository linked in this paper.**



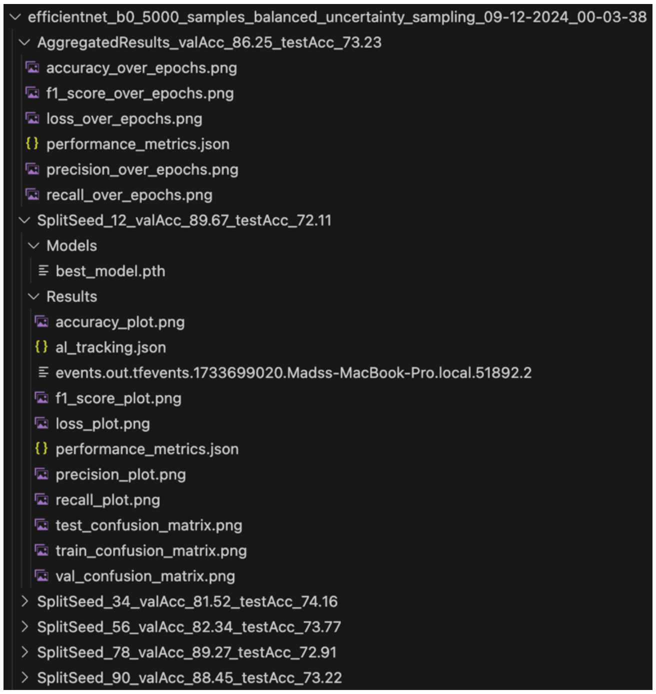
	Experiment #	Dataset %	Distribution	Sampling Technique	CLIP Initialization	Rebalanced Weights	# Split Seeds Used	Total Samples	Labels Used	Training Time (sec)	Avg Train Accuracy	Avg Val Accuracy	Test Accuracy Mean	Test Accuracy Std	Test Precision Mean	Test Precision Std	Test Recall Mean	Test Recall Std	Test F1 Mean	Test F1 Std
[illegible]	1	75%	Balanced	Stratified	No	No	5	37500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	2	50%	Balanced	Stratified	No	No	5	25000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	3	40.20%	Imbalanced (8C@50%, 2C@1%)	Stratified	No	No	5	20100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	4	40.20%	Imbalanced (8C@50%, 2C@1%)	Uncertainty	No	No	5	20100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	5	40.20%	Imbalanced (8C@50%, 2C@1%)	Margin-Based	No	No	5	20100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	6	40.20%	Imbalanced (8C@50%, 2C@1%)	TypiClust	No	No	5	20100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	7	10%	Balanced	Stratified	No	No	5	5000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	8	10%	Balanced	Uncertainty	No	No	5	5000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	9	10%	Balanced	Uncertainty	Yes	No	5	5000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	10	10%	Balanced	Margin-Based	No	No	5	5000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	11	10%	Balanced	Margin-Based	Yes	No	5	5000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	12	10%	Balanced	TypiClust	No	No	5	5000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	13	10%	Balanced	TypiClust	Yes	No	5	5000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	14	8.20%	Imbalanced (8C@10%, 2C@1%)	Stratified	No	No	5	4100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	15	8.20%	Imbalanced (8C@10%, 2C@1%)	Stratified	nan	No	5	4100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	16	8.20%	Imbalanced (8C@10%, 2C@1%)	Uncertainty	No	No	5	4100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	17	8.20%	Imbalanced (8C@10%, 2C@1%)	Uncertainty	nan	[illegible]	5	4100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	18	8.20%	Imbalanced (8C@10%, 2C@1%)	Margin-Based	No	No	5	4100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	19	8.20%	Imbalanced (8C@10%, 2C@1%)	Margin-Based	nan	[illegible]	5	4100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	20	8.20%	Imbalanced (8C@10%, 2C@1%)	TypiClust	No	No	5	4100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	21	8.20%	Imbalanced (8C@10%, 2C@1%)	TypiClust	nan	[illegible]	5	4100	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	22	5%	Balanced	Stratified	No	No	5	2500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	23	5%	Balanced	Uncertainty	No	No	5	2500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	24	5%	Balanced	Margin-Based	No	No	5	2500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	25	5%	Balanced	TypiClust	No	No	5	2500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	26	3%	Balanced	Stratified	No	No	5	1500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	27	3%	Balanced	Uncertainty	No	No	5	1500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	28	3%	Balanced	Uncertainty	Yes	No	5	1500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	29	3%	Balanced	Margin-Based	No	No	5	1500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	30	3%	Balanced	Margin-Based	Yes	No	5	1500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	31	3%	Balanced	TypiClust	No	No	5	1500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	32	3%	Balanced	TypiClust	Yes	No	5	1500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	33	2.60%	Imbalanced (8C@3%, 2C@1%)	Stratified	No	No	5	1300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	34	2.60%	Imbalanced (8C@3%, 2C@1%)	Stratified	nan	[illegible]	5	1300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	35	2.60%	Imbalanced (8C@3%, 2C@1%)	Uncertainty	No	No	5	1300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	36	2.60%	Imbalanced (8C@3%, 2C@1%)	Uncertainty	nan	[illegible]	5	1300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	37	2.60%	Imbalanced (8C@3%, 2C@1%)	Margin-Based	No	No	5	1300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	38	2.60%	Imbalanced (8C@3%, 2C@1%)	Margin-Based	nan	[illegible]	5	1300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	39	2.60%	Imbalanced (8C@3%, 2C@1%)	TypiClust	No	No	5	1300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	40	2.60%	Imbalanced (8C@3%, 2C@1%)	TypiClust	nan	[illegible]	5	1300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	41	1%	Balanced	Stratified	No	No	5	500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	42	1%	Balanced	Uncertainty	No	No	5	500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	43	1%	Balanced	Margin-Based	No	No	5	500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	44	1%	Balanced	TypiClust	No	No	5	500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Table 2: Specifications and Key Performance Metrics for Each Experiment.

7.5. Appendix E: Overview of Data Logging and Plotting

The following screenshots and plots illustrate the organization and structure of the collected data. They include the folder structure used to store experiment results, the layout of JSON files containing detailed metrics, and examples of metrics plotted by epoch, such as accuracy, F1 score, loss, and per-class confusion matrices for the CIFAR-10 dataset.

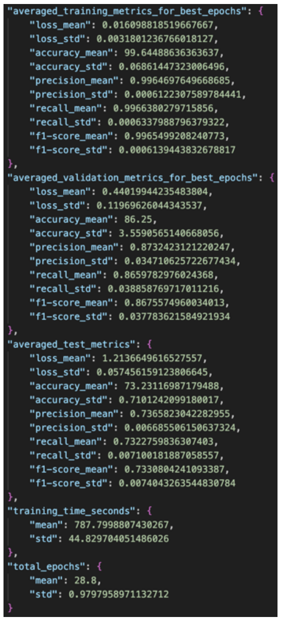
Figure 8 illustrates the folder structure used to organize the files generated for each of the 44 experiment models. It highlights the files created for storing individual results and aggregated performance metrics.



```
∨ efficientnet_b0_5000_samples_balanced_uncertainty_sampling_09-12-2024_00-03-38
  ∨ AggregatedResults_valAcc_86.25_testAcc_73.23
      accuracy_over_epochs.png
      f1_score_over_epochs.png
      loss_over_epochs.png
    {} performance_metrics.json
      precision_over_epochs.png
      recall_over_epochs.png
  ∨ SplitSeed_12_valAcc_89.67_testAcc_72.11
    ∨ Models
      ≡ best_model.pth
    ∨ Results
        accuracy_plot.png
      {} al_tracking.json
      ≡ events.out.tfevents.1733699020.Madss-MacBook-Pro.local.51892.2
        f1_score_plot.png
        loss_plot.png
      {} performance_metrics.json
        precision_plot.png
        recall_plot.png
        test_confusion_matrix.png
        train_confusion_matrix.png
        val_confusion_matrix.png
  > SplitSeed_34_valAcc_81.52_testAcc_74.16
  > SplitSeed_56_valAcc_82.34_testAcc_73.77
  > SplitSeed_78_valAcc_89.27_testAcc_72.91
  > SplitSeed_90_valAcc_88.45_testAcc_73.22
```

Figure 8: Folder Structure for Experiment data.

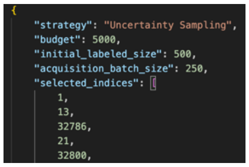
Figure 9 shows the organization of the *performance_metrics.json* file within the *AggregatedResults* folder.



```
"averaged_training_metrics_for_best_epochs": {
    "loss_mean": 0.016098818519667667,
    "loss_std": 0.003180123676601812 7,
    "accuracy_mean": 99.64488636363637,
    "accuracy_std": 0.06861447323006496,
    "precision_mean": 0.9964697649668685,
    "precision_std": 0.0006122307589784441,
    "recall_mean": 0.9966380279715856,
    "recall_std": 0.0006337988796379322,
    "f1-score_mean": 0.9965499208240773,
    "f1-score_std": 0.0006139443832678817
},
"averaged_validation_metrics_for_best_epochs": {
    "loss_mean": 0.44019944235483804,
    "loss_std": 0.11969626044343537,
    "accuracy_mean": 86.25,
    "accuracy_std": 3.5590565140668056,
    "precision_mean": 0.8732423121220247,
    "precision_std": 0.034710625722677434,
    "recall_mean": 0.8659782976024368,
    "recall_std": 0.038858769717011216,
    "f1-score_mean": 0.8675574960034013,
    "f1-score_std": 0.037783621584921934
},
"averaged_test_metrics": {
    "loss_mean": 1.2136649616527557,
    "loss_std": 0.057456159123806645,
    "accuracy_mean": 73.23116987179488,
    "accuracy_std": 0.7101242099180017,
    "precision_mean": 0.7365823042282955,
    "precision_std": 0.006685506150637324,
    "recall_mean": 0.7322759836307403,
    "recall_std": 0.007100181887058557,
    "f1-score_mean": 0.7330804241093387,
    "f1-score_std": 0.007404326354483078 4
},
"training_time_seconds": {
    "mean": 787.7998807430267,
    "std": 44.829704051486026
},
"total_epochs": {
    "mean": 28.8,
    "std": 0.9797958971132712
}
```

Figure 9: Structure of performance_metrics.json.

Figure 10 shows the organization of the *al_tracking.json* file found within each of the *SplitSeed* folders.



```
{
    "strategy": "Uncertainty Sampling",
    "budget": 5000,
    "initial_labeled_size": 500,
    "acquisition_batch_size": 250,
    "selected_indices": [
        1,
        13,
        32786,
        21,
        32800,
```

Figure 10: Structure of al_tracking.json.

For each SplitSeed and AggregatedResults folder, the performance metrics loss, accuracy, precision, recall and F1 score are plottet by epoch.

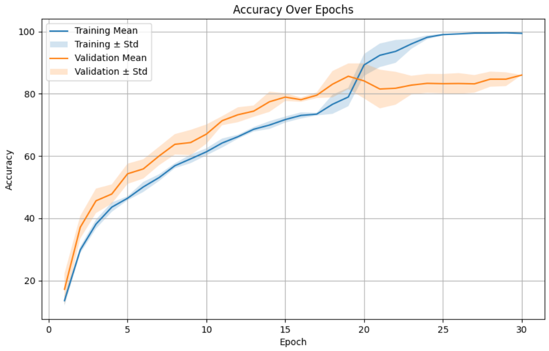
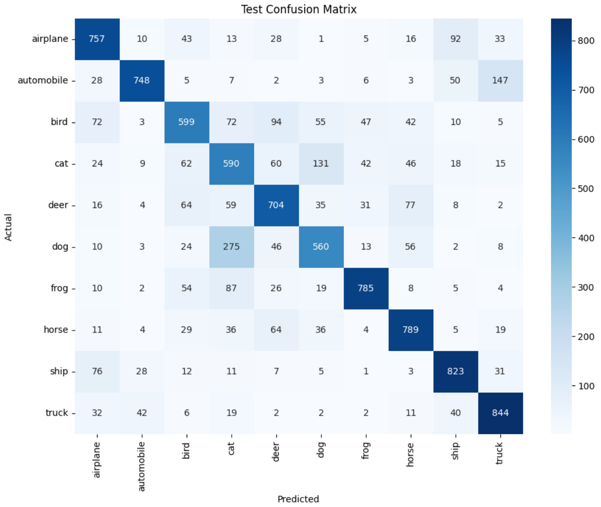


Figure 11: Example plot showing accuracy by epoch for one of the models, based on data from the AggregatedResults folder.

A per-class confusion matrix is plotted for each model from each *SplitSeed* folder.



	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	757	10	43	13	28	1	5	16	92	33
automobile	28	748	5	7	2	3	6	3	50	147
bird	72	3	599	72	94	55	47	42	10	5
cat	24	9	62	590	60	131	42	46	18	15
deer	16	4	64	59	704	35	31	77	8	2
dog	10	3	24	275	46	560	13	56	2	8
frog	10	2	54	87	26	19	785	8	5	4
horse	11	4	29	36	64	36	4	789	5	19
ship	76	28	12	11	7	5	1	3	823	31
truck	32	42	6	19	2	2	2	11	40	844

Figure 12: Example plot showing a per-class confusion matrix, generated for each model from data in the SplitSeed folders.