3LC x AWS Hackathon - Final Write-Up

Chihuahua vs. Muffin - A Data-Centric Binary Image Classification Pipeline Using 3LC

1. Introduction

This project implements a binary image classification system for distinguishing between Chihuahuas and Muffins using a data-centric AI workflow powered by 3LC. Instead of focusing on model complexity, the goal was to steadily improve classification performance by improving data quality, label accuracy, and dataset structure through iterative refinement.

The workflow leverages 3LC's capabilities - including embeddings, confidence-based filtering, misclassification analysis, clustering, and table versioning - to build a clean, consistent, and highly informative training dataset.

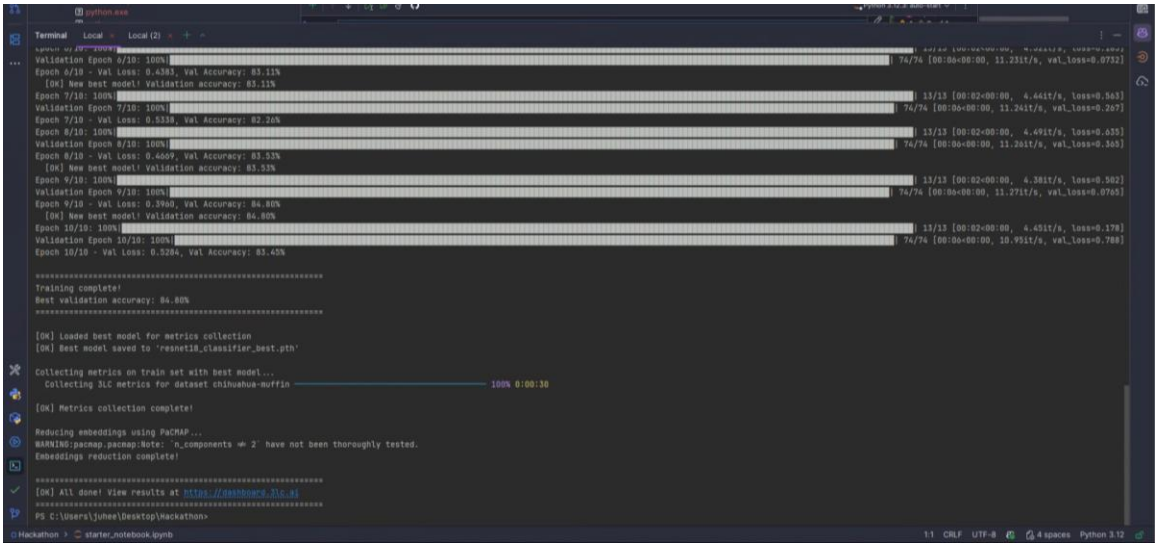
2. Baseline Setup

We began with:

- 200 labeled images (Chihuahua / Muffin)
- 4,533 unlabeled images under the undefined class
- ResNet-18 (no pretrained weights)
- Basic augmentations
- 20 training epochs, learning rate 0.0001

This created the baseline accuracy, serving as the foundation for all subsequent data-centric improvements.

Initial Accuracy: 84.80%



3. Data-Centric Iterative Loop

Our process followed the structured data-centric loop:

Train → Analyze → Fix → Retrain

Across 15 dataset versions, we consistently used model feedback, confidence metrics, embedding clusters, and misclassification reasoning to improve label quality and dataset balance.

4. Version-by-Version Dataset Improvements

Version 1 — High Confidence Auto-Labeling (>0.95)

High-confidence unlabeled samples were automatically assigned Chihuahua/Muffin labels. This expanded the dataset while maintaining precision.

Version 2 — High-Loss Sample Removal

Samples with high loss (~21.7) were identified and removed or corrected, reducing noisy and harmful examples.

Versions 3–7 — Iterative Labeling Using Confidence (>0.94 / >0.95)

Across multiple passes:

- Predicted labels were applied to unlabeled samples.
- ID-based filtering was performed.
- Outliers were re-labeled as undefined.

The dataset steadily grew with high-quality additions.

Versions 8–9 — Strict Threshold Labeling (>0.99)

Only extremely confident predictions (>0.99) were accepted. These samples were manually validated. Ambiguous or visually odd samples remained undefined.

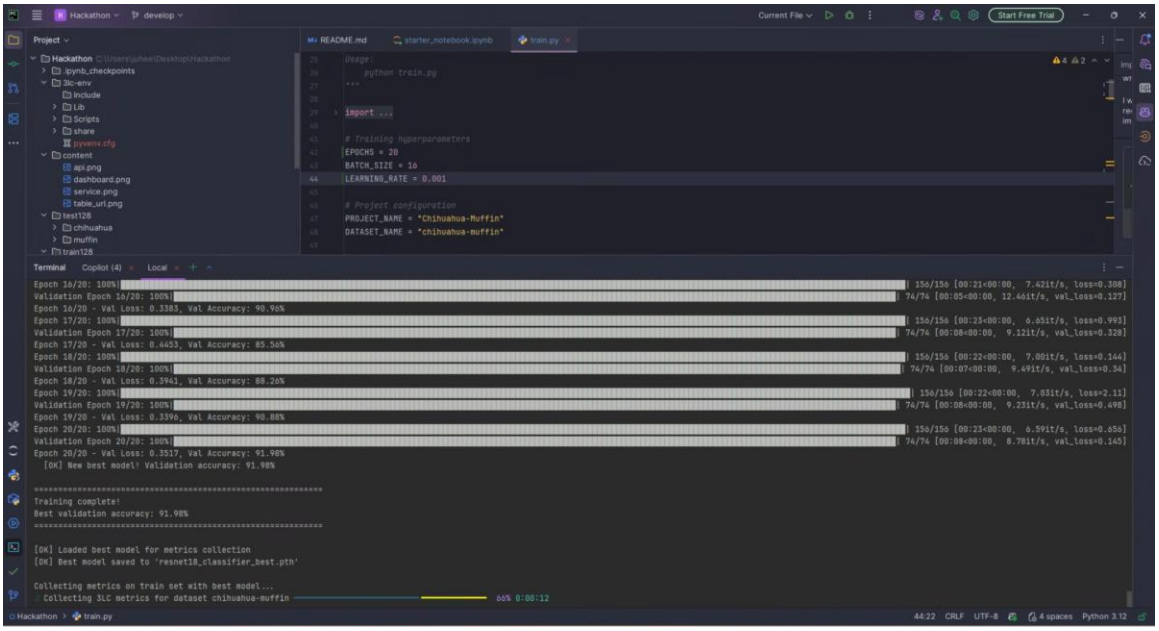
Versions 10–11 — Low Confidence Mining (<0.68 / <0.70)

- Embedding plot colored by confidence
- Low-confidence clusters were corrected
- Mislabeled and inconsistent images were removed or fixed

Hyperparameter Improvements

- Epochs increased from 20 → 50
- Learning rate increased to 0.001

Mid-Pipeline Accuracy (after Iteration 10–11): 91.98%



Version 12 — Cluster-Based Label Repairs

Low-confidence clusters were corrected, mislabels were fixed, and similar examples were grouped.

Version 13 — Reassigning Undefined Samples

Undefined samples were inspected using predicted label, confidence, and cluster location. Confident predictions were promoted to true labels.

Augmentation Upgrades

To improve generalization:
- RandomRotation
- ColorJitter
- RandomResizedCrop
- RandomHorizontalFlip

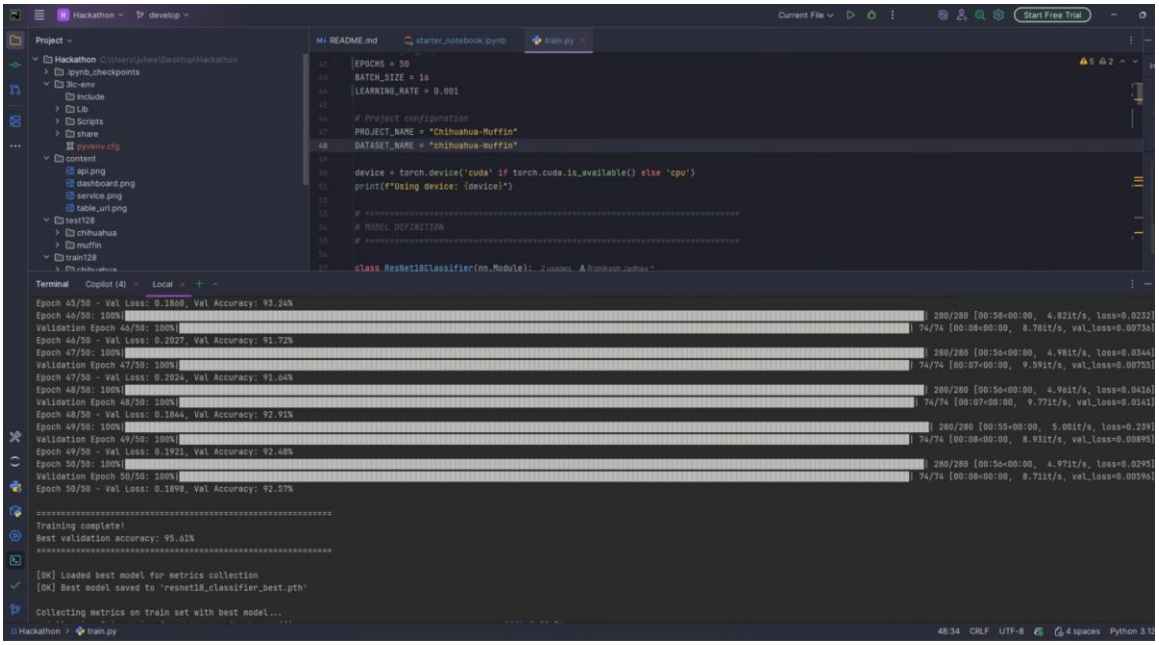
Version 14 — Outlier Removal & Fine Cluster Cleaning

Removed noisy images, incorrect groupings, and visually inconsistent samples.

Version 15 — Final Dataset Cleanup + 50 Epoch Training

A final sweep ensured all labels were correct. The final model was trained with:
- 50 epochs
- Strong augmentations
- Dynamic learning rate scheduling

Final Accuracy: 95.61%



5. Challenges & Learnings

- The biggest challenge was working with a very small labeled dataset and a large pool of unlabeled images.
- Many unlabeled images looked extremely similar, which made confidence-based labeling tricky at times.
- The model initially overfitted because the dataset was not clean, so improving labels became more important than tuning the model.
- Visualizing embeddings in 3LC helped clearly identify mislabeled samples and confusing clusters.
- A key learning was that data quality matters more than model complexity—small corrections created noticeable accuracy improvements.

6. Future Improvements

- Explore semi-supervised techniques like refined pseudo-labeling to improve confidence in difficult regions.
- Build a small manually curated validation set for more reliable accuracy tracking.
- Experiment with additional augmentation strategies for more robustness.
- Extend this workflow to multi-class problems or real-world datasets.
- Automate parts of the 3LC labeling workflow to speed up future iterations.

7. Conclusion

This project demonstrates a complete data-centric AI workflow powered by 3LC, showing how dataset refinement, not increased model complexity is the key driver of performance gains. Through iterative labeling, noise reduction, confidence analysis, clustering, and misclassification review, a robust dataset was created that enabled strong performance from a simple ResNet-18 classifier.