AC 297R: Mosaic ML Milestone 1

Alex Leonardi, Chris Gilmer-Hill, Xingyu Liu, Lu Yu

IACS INSTITUTE FOR APPLIED COMPUTATIONAL SCIENCE AT HARVARD UNIVERSITY

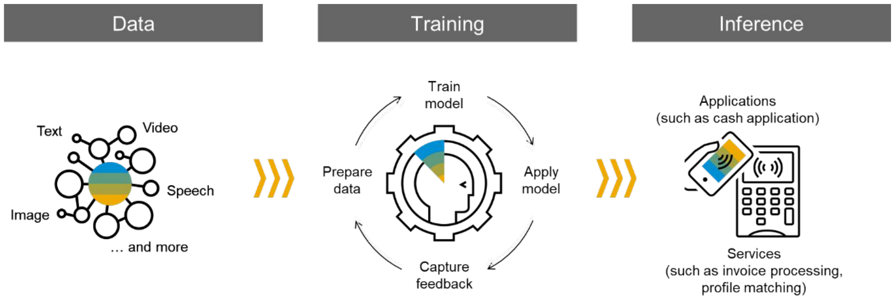
mosaicML

Table of Contents

Background

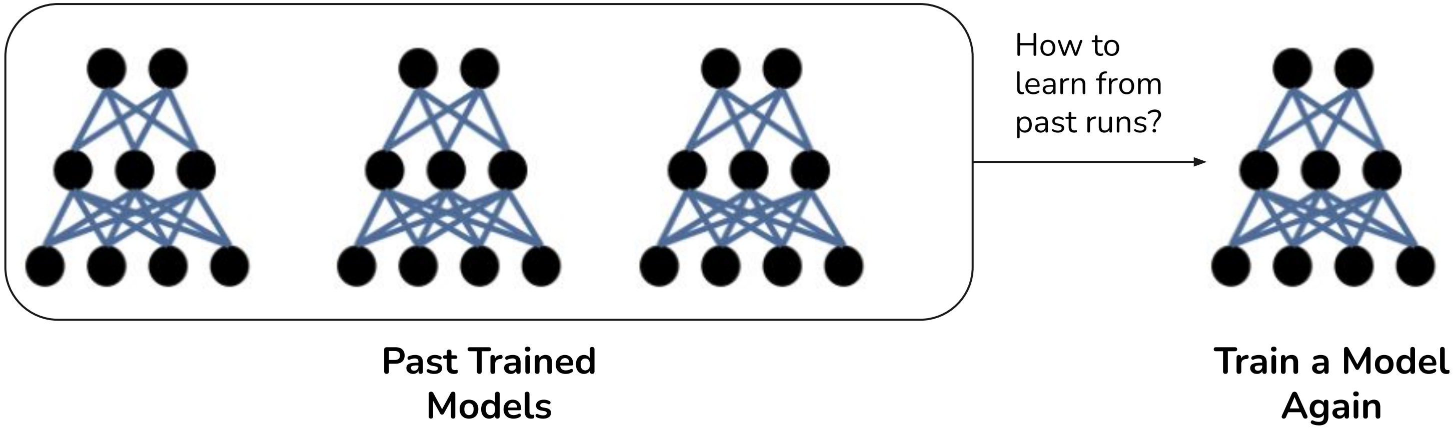
Motivation

- In ML research, models are often trained using independent runs
- Models can be loaded and training resumed from the last epoch, but this approach does not generalize to different reasons why people might want to retrain models (e.g., tuning hyperparameters, etc.)
- The result is lots of computation that isn't put to use



Problem Statement

How can we reuse the computation that was invested in training our initial models to make training future models better?



Scope of Work

I. Build Our Harness (Milestone 1)
   A. Set up AWS and other relevant resources to host our models, data, and computation
   B. Design a codebase that allows us to run experiments throughout the course of our project, scaling to different models and datasets we may eventually use
   C. Test this harness on a simple problem: training ResNet56 on CIFAR-100
II. Test Our Methods (Milestone 2)
   A. Begin experimenting with various methods to reuse computation across multiple runs, with guidance from partners at MosaicML
   B. Visualize and measure performance against common benchmarks
III. Compile Our Results (Milestone 3)
   A. Build upon initial results to finalize findings for project
   B. Compile findings into research paper / final report

Learning Goals

- Building flexible model-training infrastructure from scratch
- Understanding and implementing cutting-edge model-training techniques
- Designing experiments to tackle an open-ended research topic
- Working alongside industry partners to achieve mutually satisfactory results

Our Team & Infrastructure

Our Team

Project Partner:

- Jonathan Frankle: Chief Scientist at MosaicML

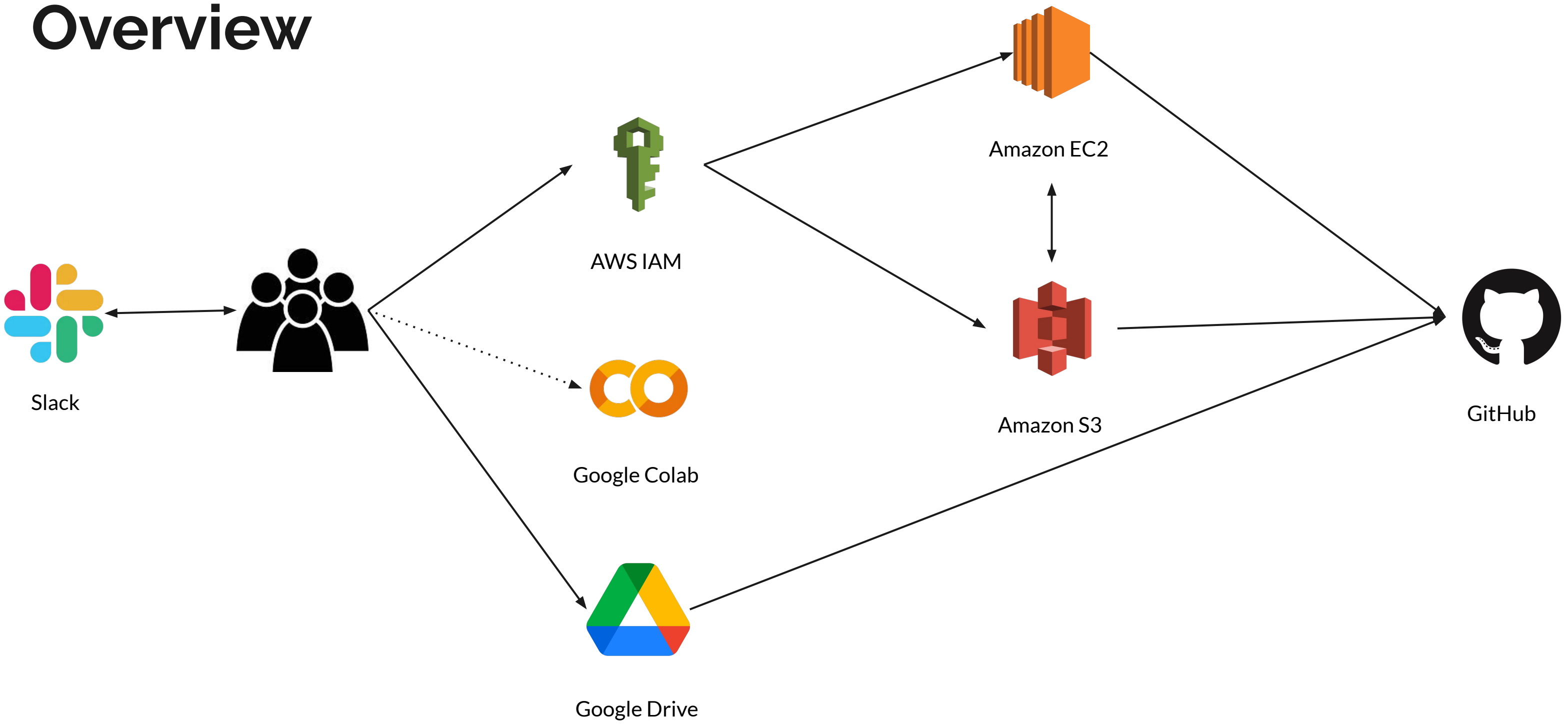
Our Team:

- Alex Leonardi: AB/SM CSE student
- Chris Gilmer-Hill: AB/SM CSE student
- Xingyu Liu: G2 Data Science student
- Lu Yu: G1 CSE student

Infrastructure

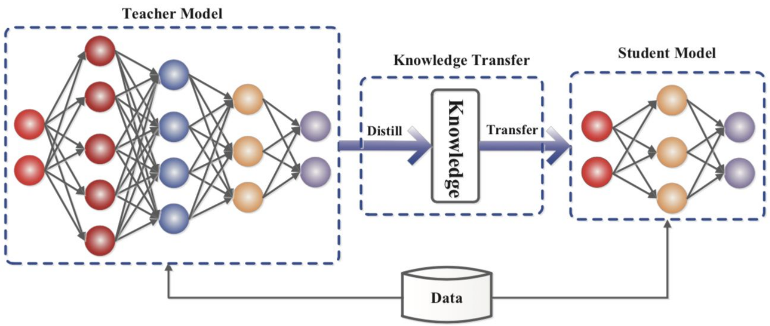
- **Storage and Computation:** AWS
  - S3 for model/dataset storage
  - EC2 for compute instance
  - Shared account with IAM login for each user already set up
- **Codebase:** GitHub, Google Colab
  - Repository for harness used to train models
  - GitHub Organization created in the event that additional repositories are needed
  - Google Colab Pro temporarily used while waiting on AWS credits
- **File-Sharing:** Google Drive
- **Communication:** Slack

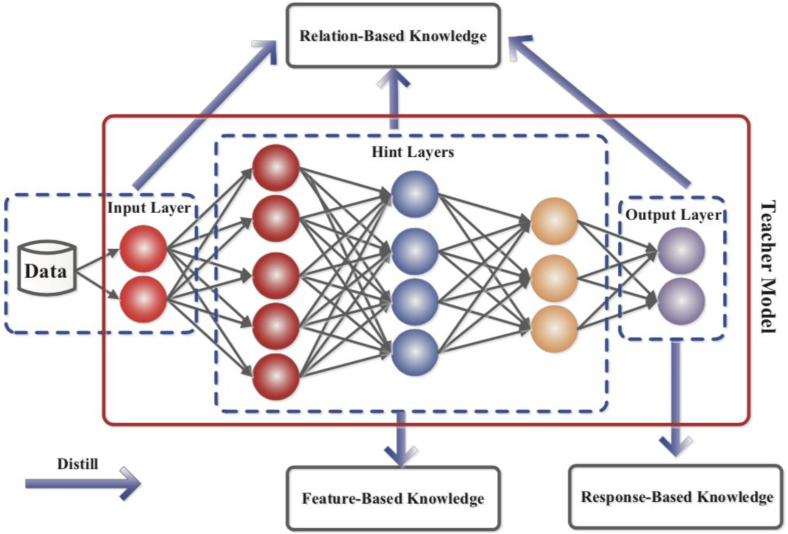
Overview


Slack
AWS IAM
Google Colab
Google Drive
Amazon EC2
Amazon S3
GitHub

Relevant Knowledge and Project Ideas

Knowledge Distillation





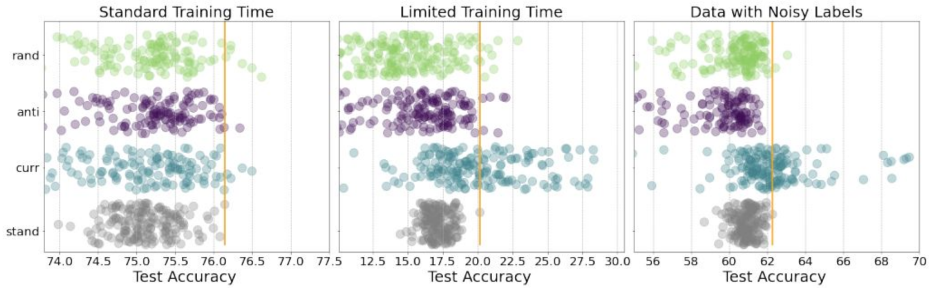
Gou et al 2021

KD + MixUp

- MixUp:
  - Data augmentation via convex linear combination (Zhang et al 2017)
  - Alternative application: interpolation between model iterations
- Proposal: **Use MixUp to generate a linear combination of model outputs across model iterations as distillation of Response-Based Knowledge**

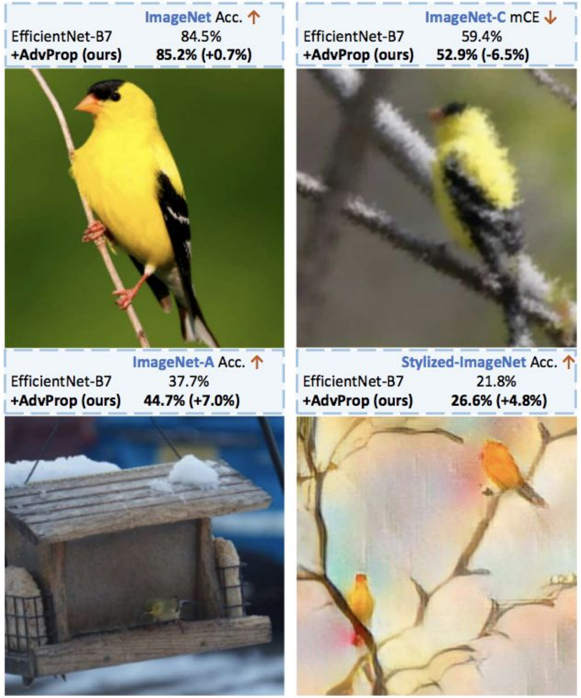
History-Informed Difficulty Scaling

- Adapted from NLP: [Sequence Length Warmup]
- GraNd score: generalized measure of "difficulty"/"importance" (Paul et al 2021)
- Curricula: sort inputs by difficulty - especially useful when training time is limited and labels are noisy/imperfect
- Proposal: **use GraNd scores from past training iterations to predict and scale importance of examples in "curricula" for future training runs**



Adversarial Recycling

- Evidence from Image Nets: in training, adversarial examples improve model accuracy overall (Xie et al 2020)
- Specifically improved in cases of “distribution mismatch”
- Proposal: **leverage adversarial examples from past model iterations in future training iterations to improve model robustness**



Works Cited for Relevant Knowledge:

1. Gou, J., Yu, B., Maybank, S.J. and Tao, D., 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, *129*(6), pp.1789-1819.
2. Paul, M., Ganguli, S. and Dziugaite, G.K., 2021. Deep Learning on a Data Diet: Finding Important Examples Early in Training. *Advances in Neural Information Processing Systems*, *34*.
3. Wu, X., Dyer, E. and Neyshabur, B., 2020. When do curricula work?. *arXiv preprint arXiv:2012.03107*.
4. Xie, C., Tan, M., Gong, B., Wang, J., Yuille, A.L. and Le, Q.V., 2020. Adversarial examples improve image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 819-828).
5. Zhang, H., Cisse, M., Dauphin, Y.N. and Lopez-Paz, D., 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Current Progress

EDA

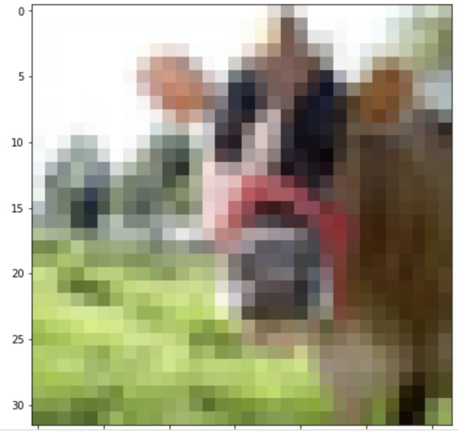
- Project Focus is model **not data**
- Current Dataset: CIFAR-100
  - 100 different classes
  - Each has 500 images
- Future Dataset
  - ImageNet
  - NLP datasets, if possible

```python
train_data = CIFAR100(root=data_dir)

fig, ax = plt.subplots(figsize=(7.5,7.5))

for image,label in train_data:
    image = np.array(image)
    print("Image shape: ",image.shape)
    ax.imshow(image)
    print("Label: ", label)
    break
```

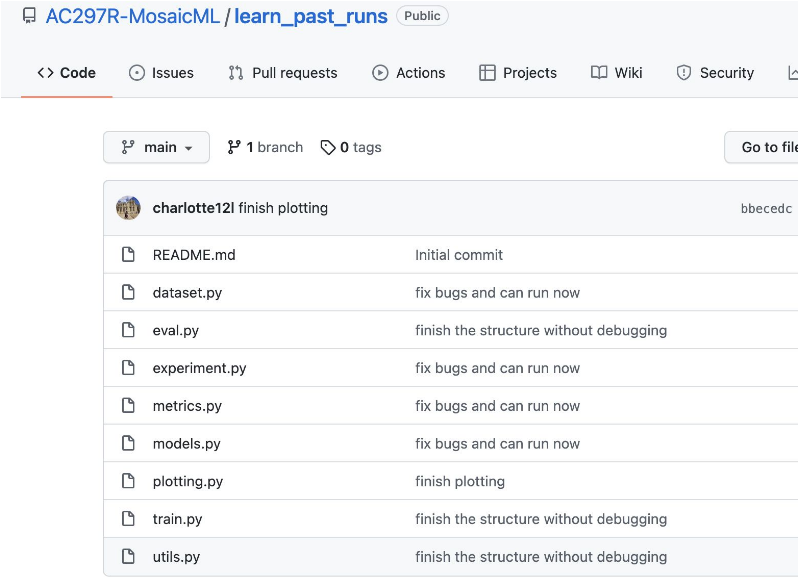
```
Image shape:  (32, 32, 3)
Label:  19
```



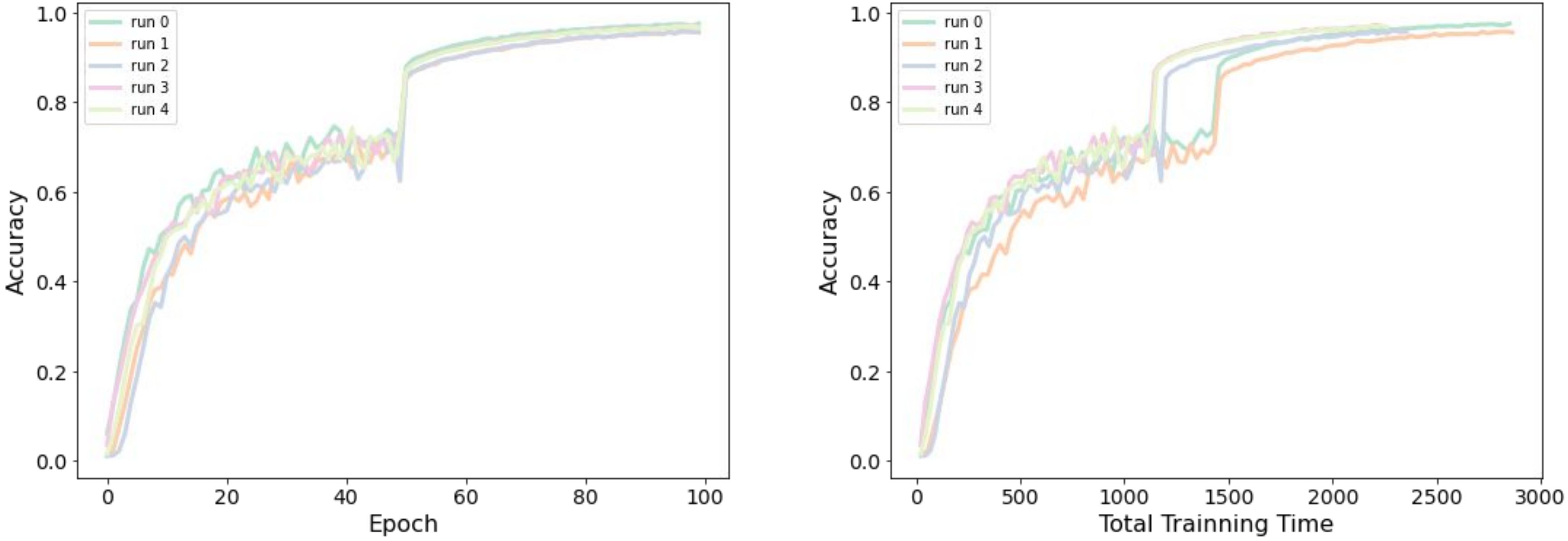
Harness Design

- **Requirements:**
  - Multiple data-sets/models
  - Control hyperparameters, etc.

- **Files:**
  - experiment.py: highest-level
    - Highest-level
    - Run with command-line arguments
  - train.py: reference/train models
    - Model training
    - Pass model and dataset to ensure reusability
  - eval.py: evaluate models
  - models.py: contains many models
  - dataset.py: dataset and pre-processing
  - metrics.py: evaluation metrics
  - plotting.py: graphing utilities



Baseline: 5 separate runs of ResNet56 for CIFAR100



Q&A