Land Cover Classification in the Amazon

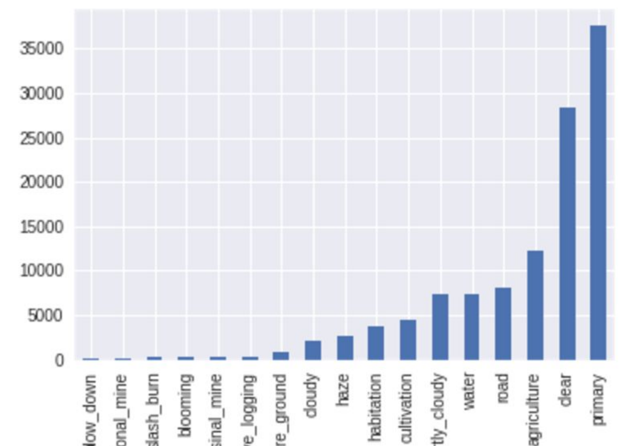
Zachary Maurer (zmaurer), Shloka Desai (shloka), Tanuj Thapliyal (tanuj)

INTRODUCTION

- The Amazon Rainforest is subject to aggressive deforestation. Automatically identifying areas of human encroachment could aid in conservation.
- We built a family of neural network classifiers which input a satellite image and output 17 distinct labels per image.
- We achieved an F2 score of 89% using transfer learning (Resnet) on a labeled dataset.
- Models were implemented in PyTorch and run on a Google Cloud GPU instance.

DATASET

- Data from Kaggle's "Planet: Understanding the Amazon from Space" challenge [1]
- The dataset consists of over 110K 256 x 256 image tiles labelled with at least one of 17 classes such as "primary rainforest", "water", "habitation", "cultivation", and "cloudy"
- 40K images for training; 60K images for test



- The most common label is "primary rainforest", followed by "clear". Images can have only one for four weather labels "haze", "clear", "cloudy","partly cloudy".
- Dataset was augmented during training with random jitter, cropping, scaling and transposing.

MODELS

Baseline

- **XGBoost**
  - 18 features per image - mean, std, median, min/max, skew, kurtosis per channel (RGB)

Neural Models

- **Baseline CNN**
  - 3 layer sequential Conv-net + 2 FC layers
  - Adam Optimizer (β_1=0.9, β_2=0.999)
- **U-Net [2]**
  - Recursive convolutional neural network.
  - Double-conv blocks that are successively downsampled and then recombined through upsampling.
  - Observed to perform well on image segmentation.
  - SGD with Momentum(0.9)

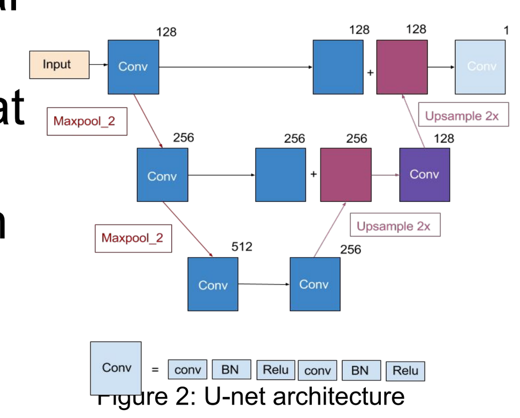


Figure 2: U-net architecture

- **Pyramid Net[5,6]**
  - Similar recursive structure to U-Net
  - Uses residual connections in addition to max pooling on the recursive "down".
  - All spatial levels are fed to aggregation and prediction layer.
  - SGD with Momentum(0.9)
- Above neural models trained with multi-label soft-margin loss and a sigmoidal predictions.

$$\text{loss}(x,y) = -\frac{1}{n}\sum_i^n (y_i \log(\frac{exp(x_i)}{(1+exp(x_i)}) + (1-y_i)\log(\frac{1}{1+exp(x_i)}))$$

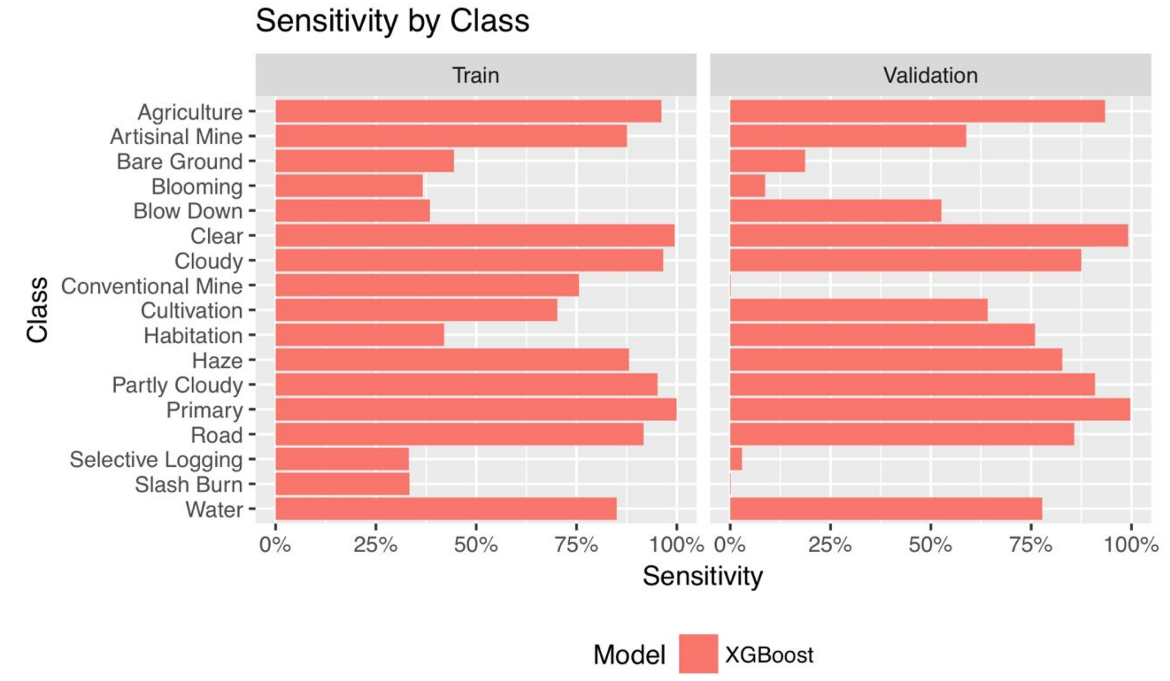
  - VGG16 [3], trained last two FC layers
  - ResNet50 [4], trained last FC layer
  - ResNet50, trained last Conv block + FC
  - ResNet50, last two Conv blocks + FC

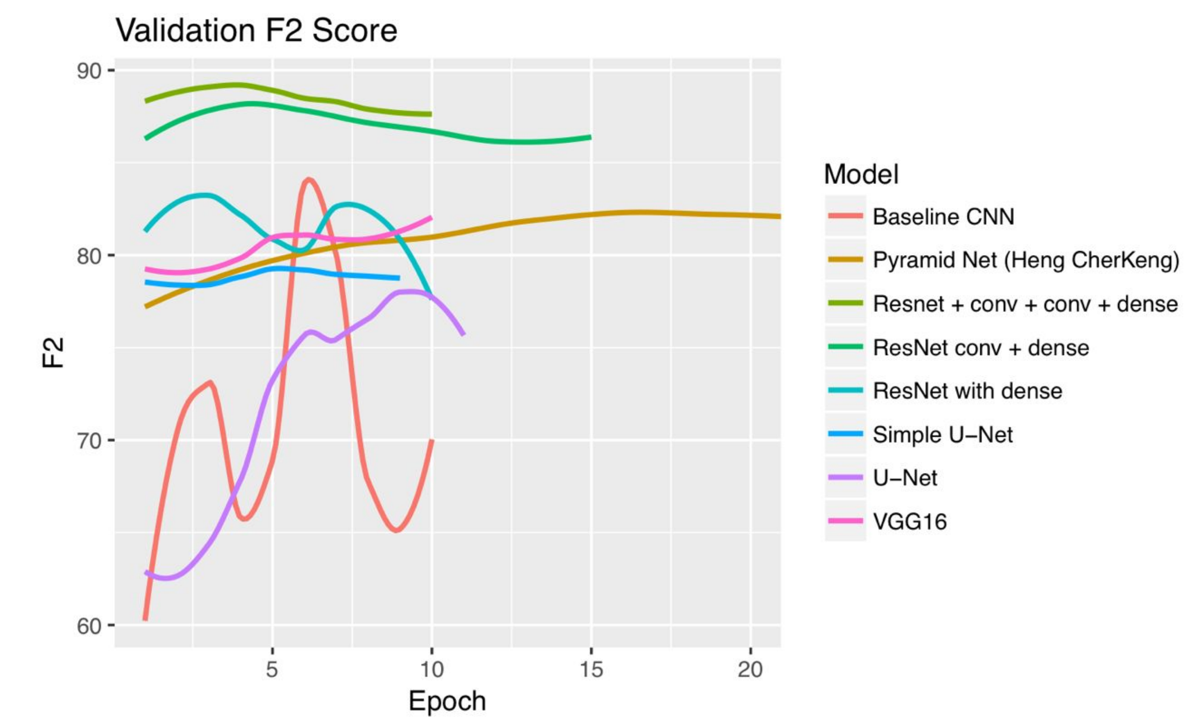
Model	Number of Parameters
XGBoost	Depth: 5, Estimators: 100
Baseline CNN	14.8M
U-Net	8.6M
Simple U-Net	13.3M
Pyramid Net (Heng CherKeng)	3.7M
ResNet (retrained Dense)	35K
ResNet (retrain Conv + Dense)	14.9M
ResNet (Conv + Conv + Dense)	22M
VGG16	70K

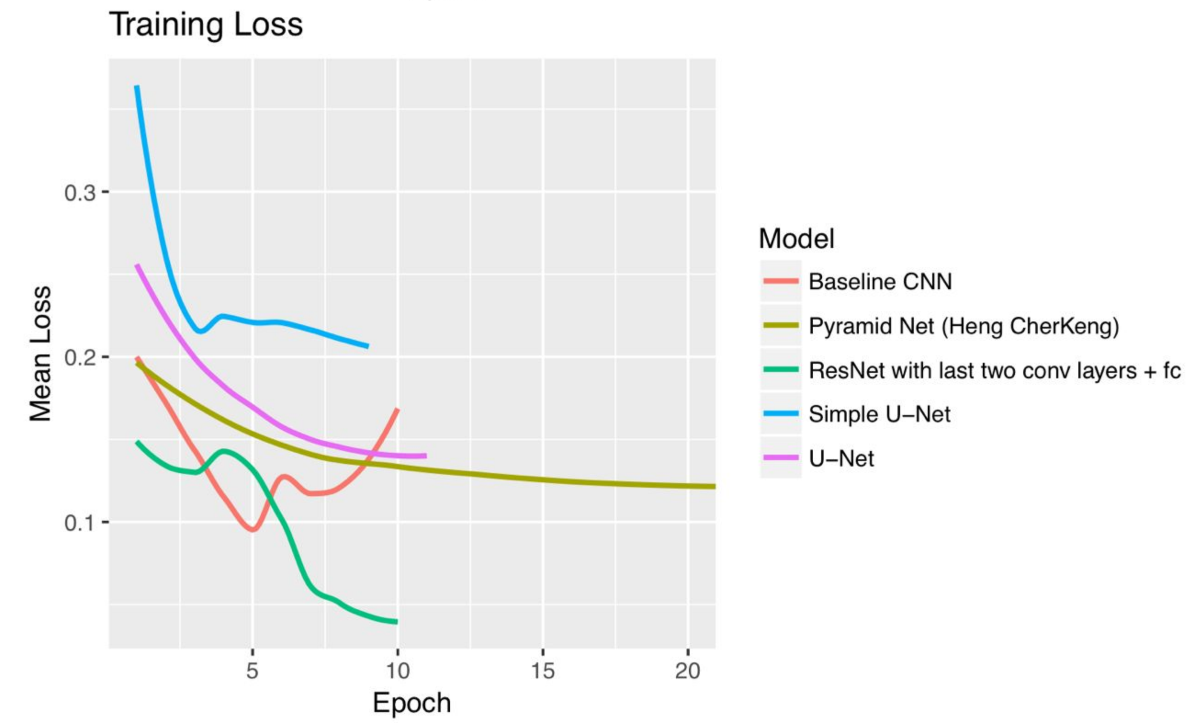
RESULTS

Validation Set Performance

Model	F2-Score	Recall	Exact Match
ResNet (Conv + Conv + Dense)	89.24%	95.39%	58.50%
ResNet (retrain Conv + Dense)	88.48%	95.16%	56.69%
Baseline CNN	85.35%	93.89%	48.74%
ResNet (retrained Dense)	84.81%	93.86%	48.70%
Pyramid Net (Heng CherKeng)	84.08%	94.02%	46.83%
U-Net	82.41%	93.19%	46.13%
VGG16	82.18%	92.97%	42.33%
Simple U-Net	79.61%	91.70%	37.98%
XGBoost	79.40%	92.24%	39.16%
Simple U-Net (Jacc. Loss)	74.40%	90.82%	32.00%







DISCUSSION

- Transfer learning performs the best amongst all approaches. Specifically, ResNet50 performed best.
- Recursive neural networks seem to perform well on possibly related tasks (i.e. image segmentation). However, a standard U-Net implementation was not as performant as a U-Net with modified residual connections such as Pyramid Net.
- XGBoost is a formidable baseline given the simplicity of feature extraction and training.
- The imbalanced dataset caused per-class sensitivity to vary significantly.
- Using SGD-Momentum(μ = 0.9) was crucial to lowering loss below 0.2. LR Decay with Adam halted progress for early results.
- Per-class loss weighting did not improve performance on unbalanced dataset.
- Upsampling minority classes using a Least Squares Projection caused immediate overfitting that couldn't be solved with dropout.
- Modifying the loss function to include Jaccard distance seems to have been used effectively in other cases, but requires more tuning.

FUTURE WORK

- Train multiple sub-networks that specialize for label type.
- Condition neural architectures on statistical features.
- Ensemble all trained models.
- Train models on TIF infrared channel data.
- As an experiment, feed Resnet features into XGBoost
- Train models on entire dataset and compete on Kaggle
- Continue to tune hyperparameters, new LR schedules

REFERENCES

[1] Kaggle Planet Understanding the Amazon from Space Challenge: https://www.kaggle.com/c/planet-understanding-the-amazon-from-space/
[2] Von Eicken, Thorsten, et al. "U-Net: A user-level network interface for parallel and distributed computing." ACM SIGOPS Operating Systems Review. Vol. 29. No. 5. ACM, 1995.
[3] Very Deep Convolutional Networks for Large-Scale Image Recognition K. Simonyan, A. Zisserman. arXiv:1409.1556
[4] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.
[5] https://github.com/hengck23-kaggle/kaggle_forest_2017
[6]https://www.kaggle.com/c/planet-understanding-the-amazon-from-space/discussion/32402