Land-Use and Deforestation in the Brazilian Amazon

Using Satellite Imagery and Deep Learning for Environmental Conservation

Cameron Bronstein

DSI Capstone

Agenda

- Introduction
- Data Collection with Planet API
- Exploratory Data Analysis & Image Preprocessing
- Model Building & Evaluation
- Recommendations & Future Steps

Introduction

- The Amazon rainforest is an eco-climate system vital to regulation of global climate, via oxygen production and carbon sequestration.
- Facing huge deforestation pressure from illegal logging and cattle ranching.
- Over 20% of the land cover has been deforested.
- Nearly 8000 square km logged from Aug 2017 - July 2018

Fishboning in Rondônia

Introduction

- Planet, a satellite imaging company, has the largest fleet of satellites in the world.
- In Brazil, Planet partners with a conservation consultancy (SCCON) to monitor land coverage in the Amazon with remote sensing, providing high resolution imaging that can distinguish human-caused from natural forest loss.
- Currently, no in place methods exist for automated classification of images into a multitude of land use and land cover categories.

Introduction

Problem Statement:

- Using Satellite Imagery from Planet, build a method to classify images by weather, land use and land cover types.
- Effective models will correctly predict images with multiple land-use types.

Applications:

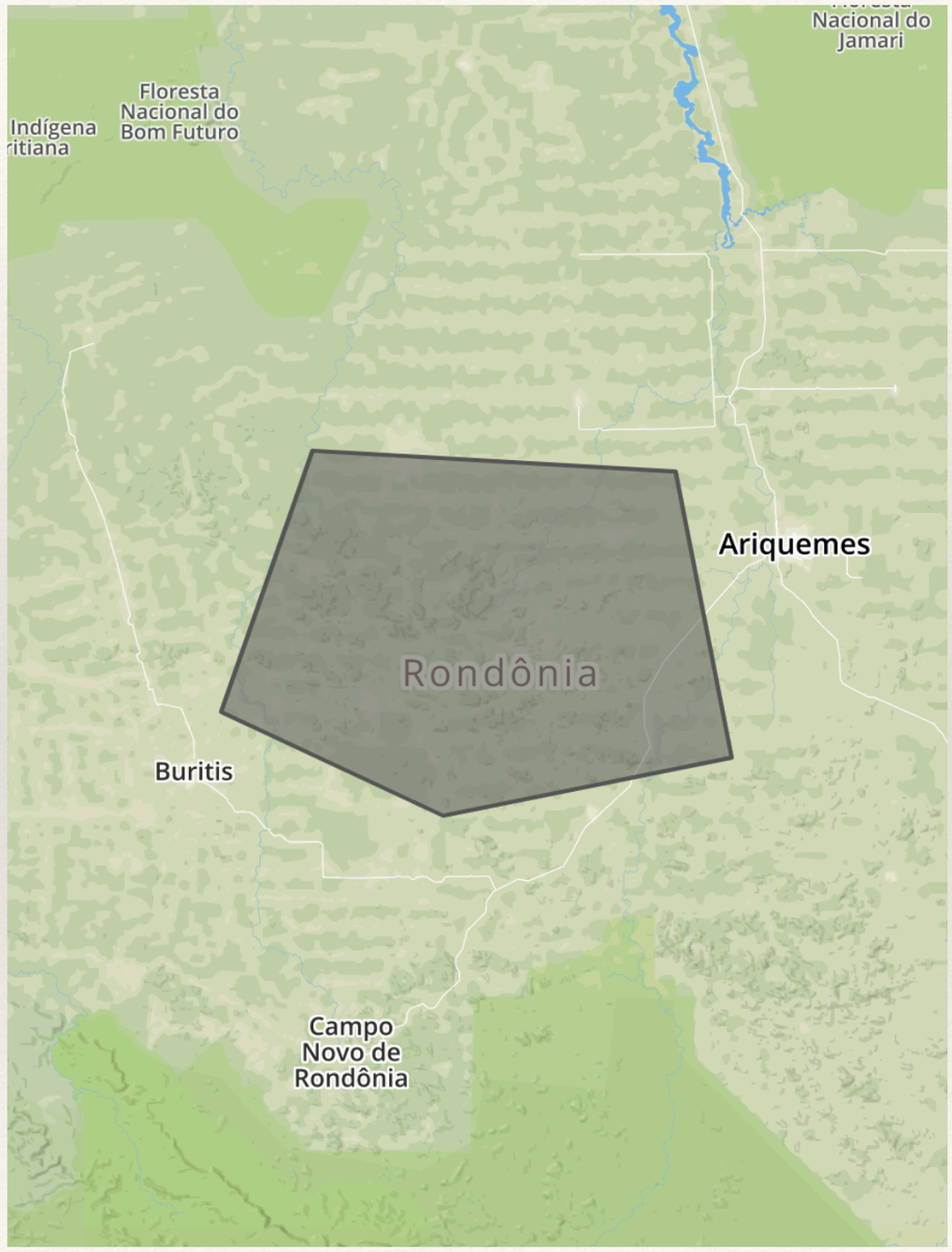
- Real-time identification to direct conservation organizations and local government to stop illegal land use activities.
- This would automate the currently tedious human-verified image labeling and data entry system.

Data Pipeline with Planet API

- To generate images, the first task is to specify and Area of Interest (AOI).
- Create compound filter: AOI and time period.
- Specify item type (Satellite type) and assets (analytic, visual, etc.)
- Different Item types have different resolutions and contain different associate metadata.

Data Pipeline

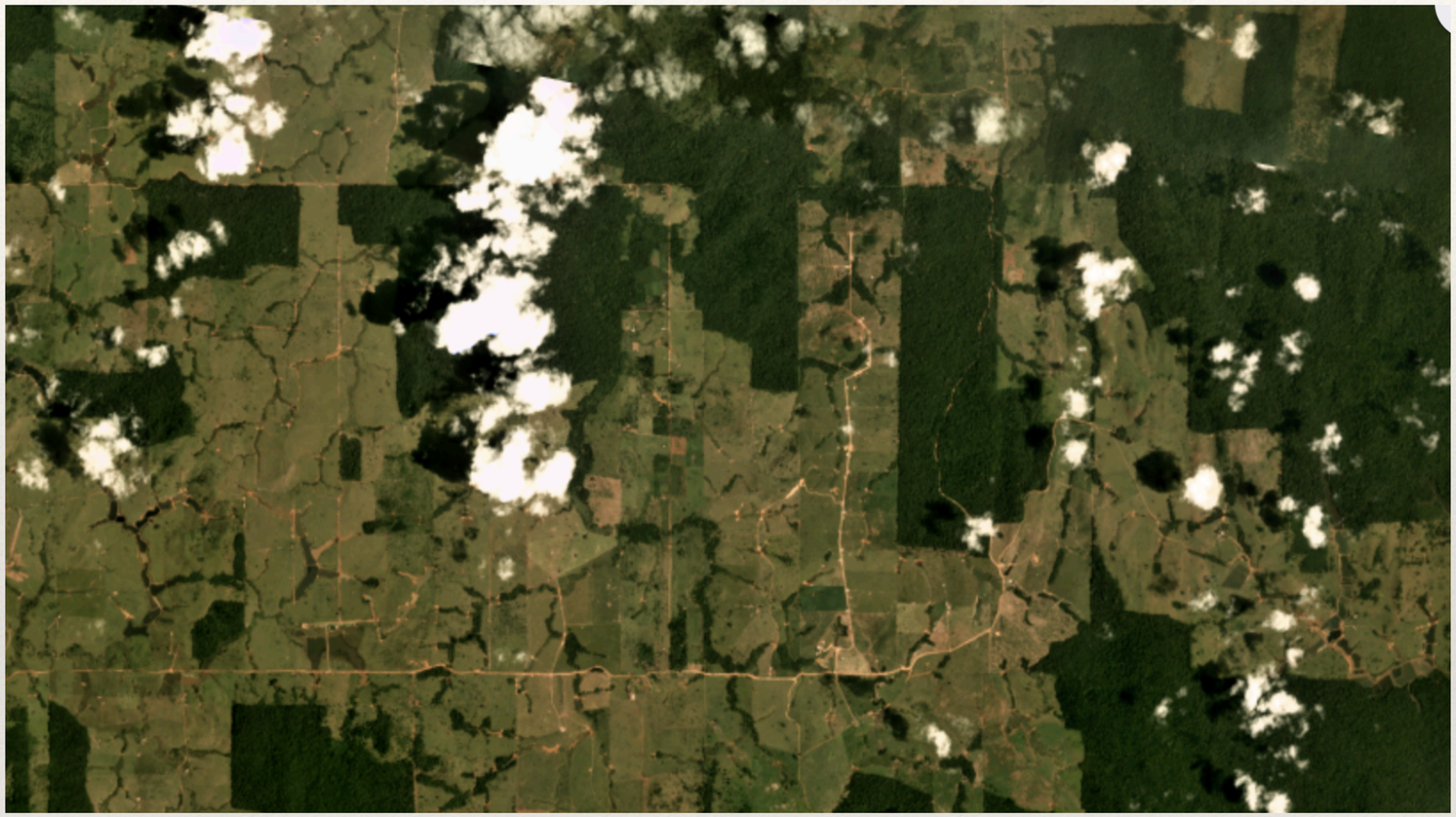
- AOIs can be really large!
- API requests include images that contain fragments that lie within the AOI.
- Images need to be augmented, rotated, cropped significantly to be processable by a neural network.



AOI in Rondônia, Brazil

Image Labels

Party cloudy - Road - Agriculture - Primary



Land Use and Land Cover Labels

Primary Forest - Road - Water - Agriculture - Habitation - Bare Ground - Cultivation - Blooming - Artisanal Mine -Selective Logging - Slash and Burn - Conventional Mine - Blow Down

Weather Labels

Clear - Haze - Cloudy - Partly Cloudy

The Data

Provided by Planet and Kaggle

Dataset

- 40,000 training images
- 60,000 test images

Image Specs

- Red: 610 - 700 nm
- Green: 500 - 590 nm
- Blue: 420 - 530 nm
- Near Infrared: 700 - 1000 nm

Resolution

- 256 x 256 Pixels
- 3 meter resolution
- 768 m^2 per image

Understanding the Labels

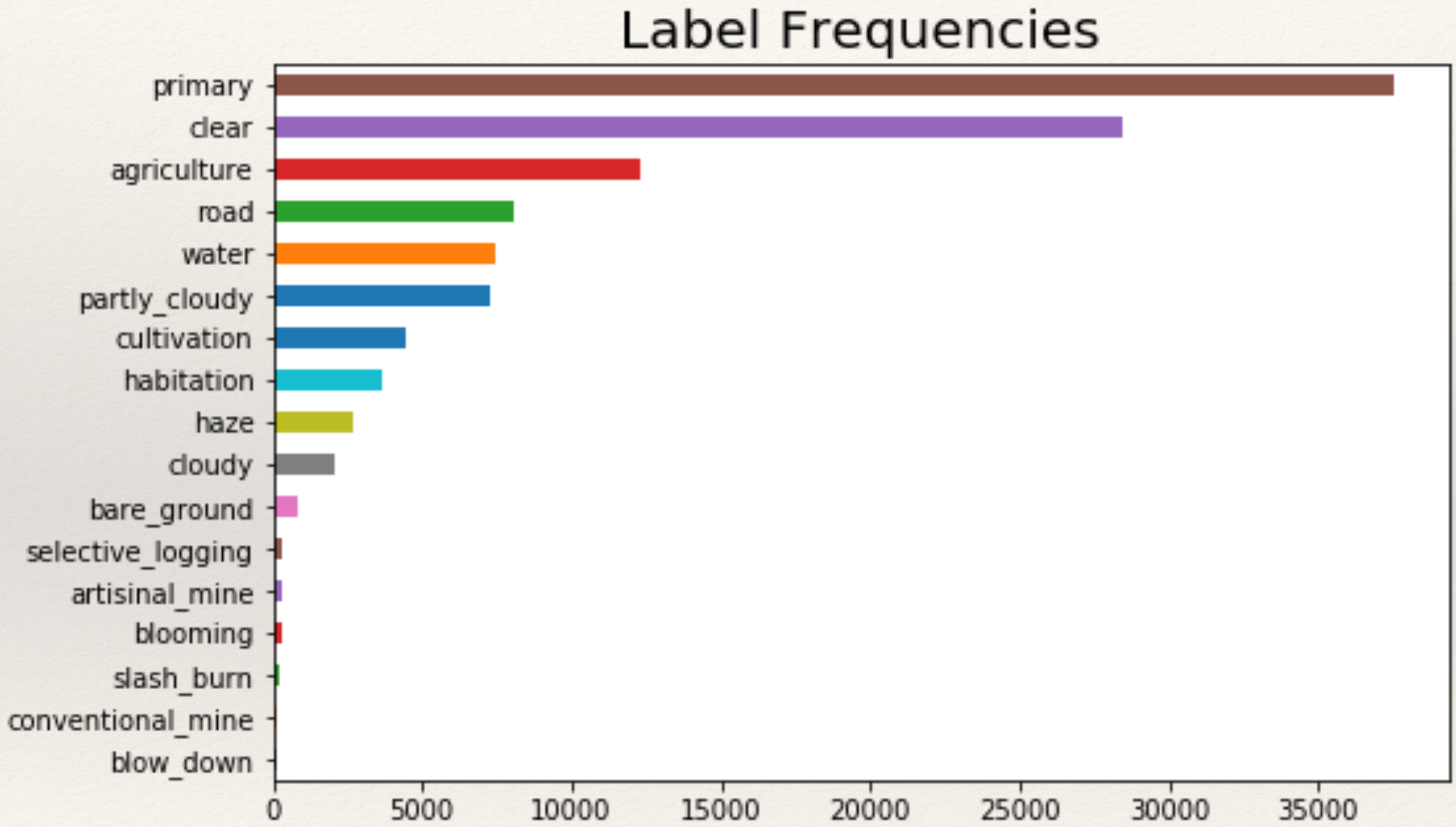
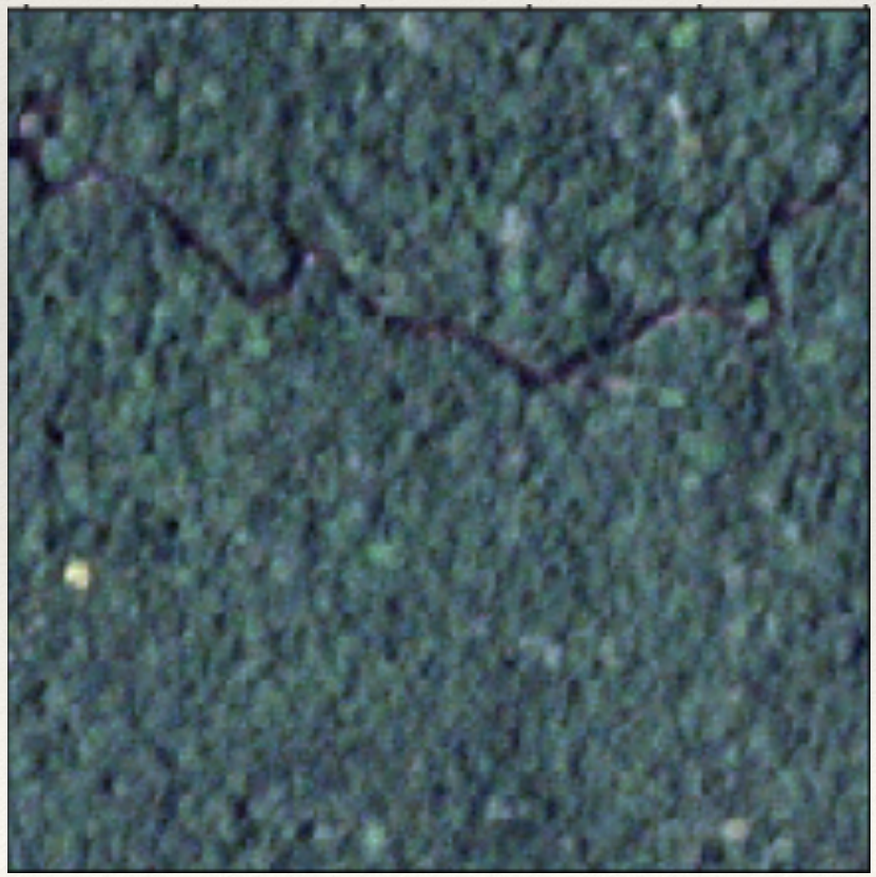


Image Pre-Processing

- Neural networks “learn” by recognizing patterns in image features. For each image, a feature is a pixel.
- However, if a certain weather or land type only appears in a certain region of an image, the neural network will not recognize the exact same type if it is in a different location in the image.

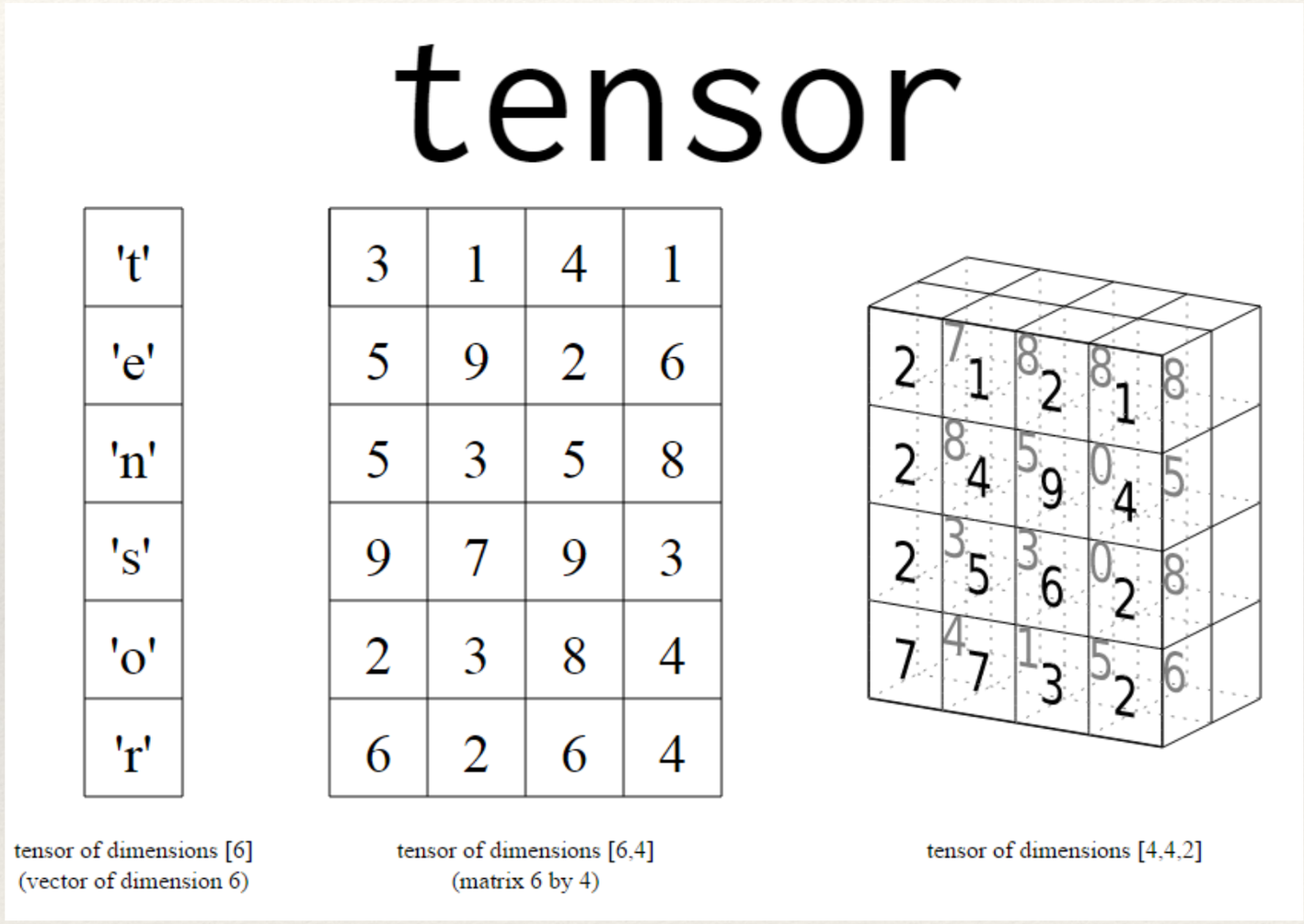
True Image



Vertical Flip

Final Image Conversion

- Inputs to the model: Image converted to tensor
- Tensor - a multi-dimensional array.
- Each image is a tensor (128 x 128 x 3).
- Only included RGB bands.



CNN Explained

- Convolutional Neural Networks (CNN) are best for image processing.
- Convolutional layers apply a filter to the image and transform its values and dimensions.
- As the network is trained, the calculations are adjusted for each transformation so that the network converges on the appropriate output for each possible prediction type.

Train - Validate - Test

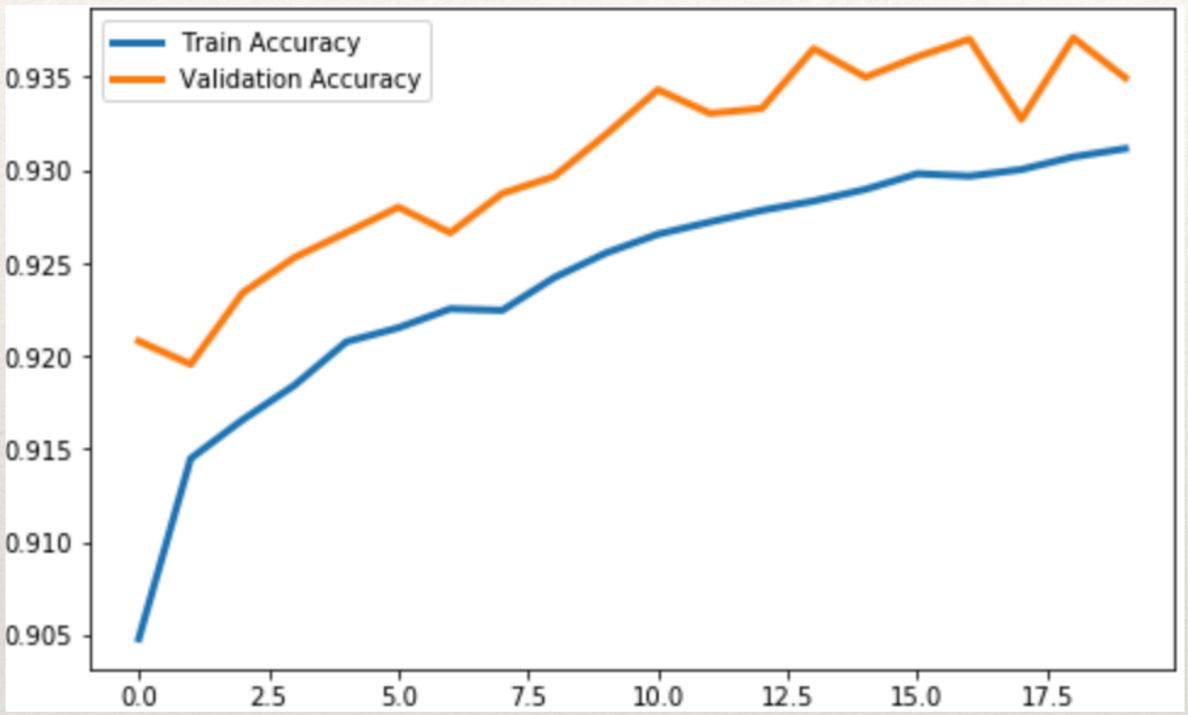
- We only have so many truly labeled images, so we have to make full use of them to train and test our model.
- Randomly splitting the labeled data into training and testing allows building and validating multiple types of models before making final predictions.
- Train the model across multiple epochs (full pass of the data through the network).

Modeling Evaluation

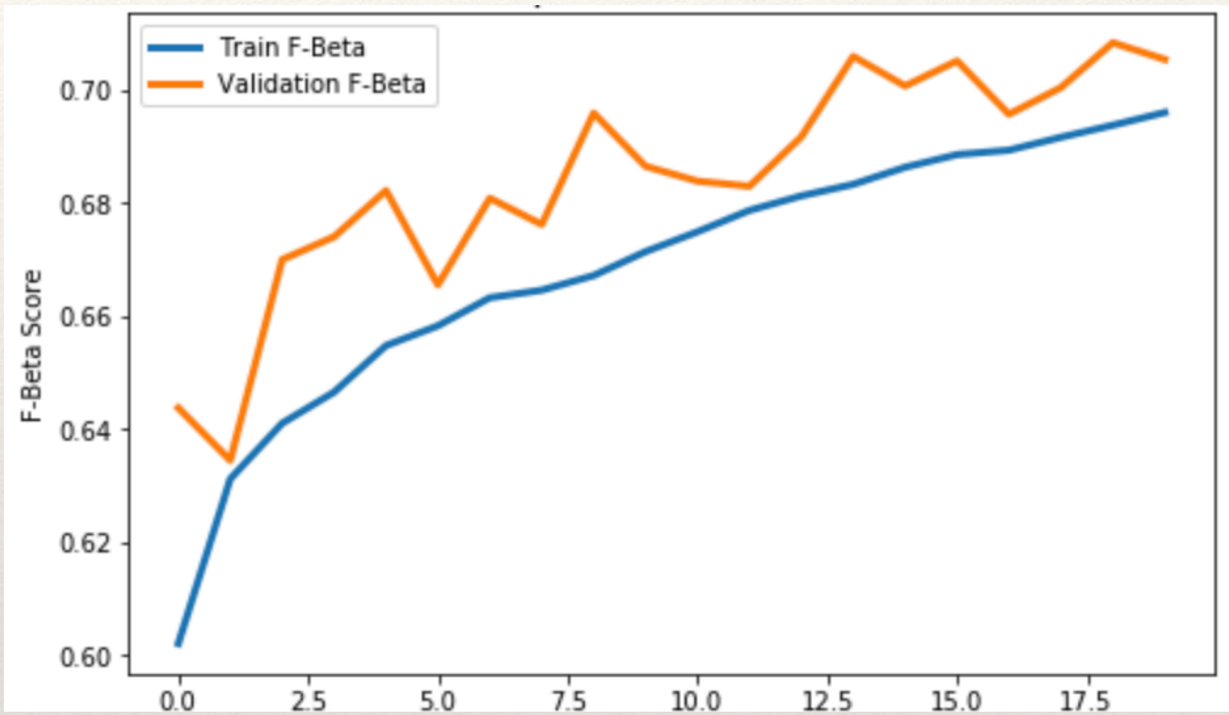
- Model outputs 17 “probabilities” – one for each label category.
- Thresholds allow us to convert to binary values.
- If probability > threshold: prediction = 1
- How do we determine the right thresholds?

Model Results

Accuracy



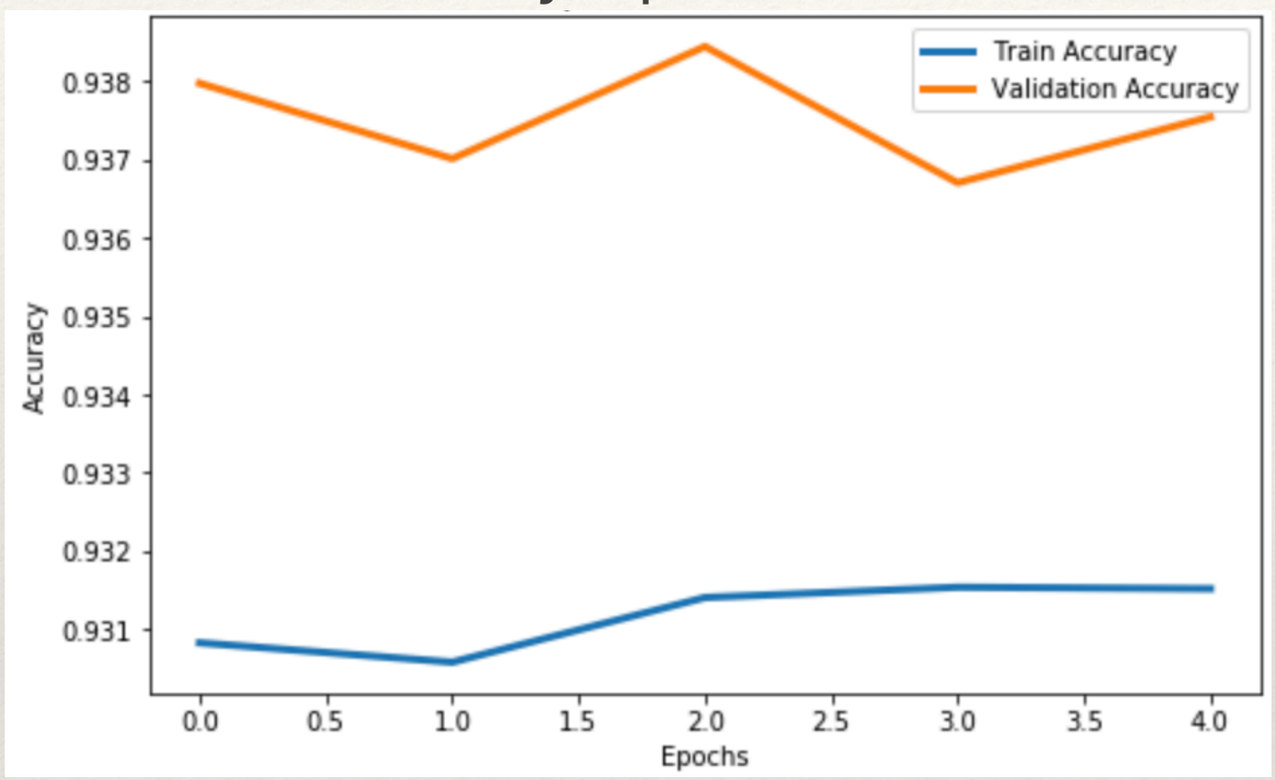
F-Beta Score



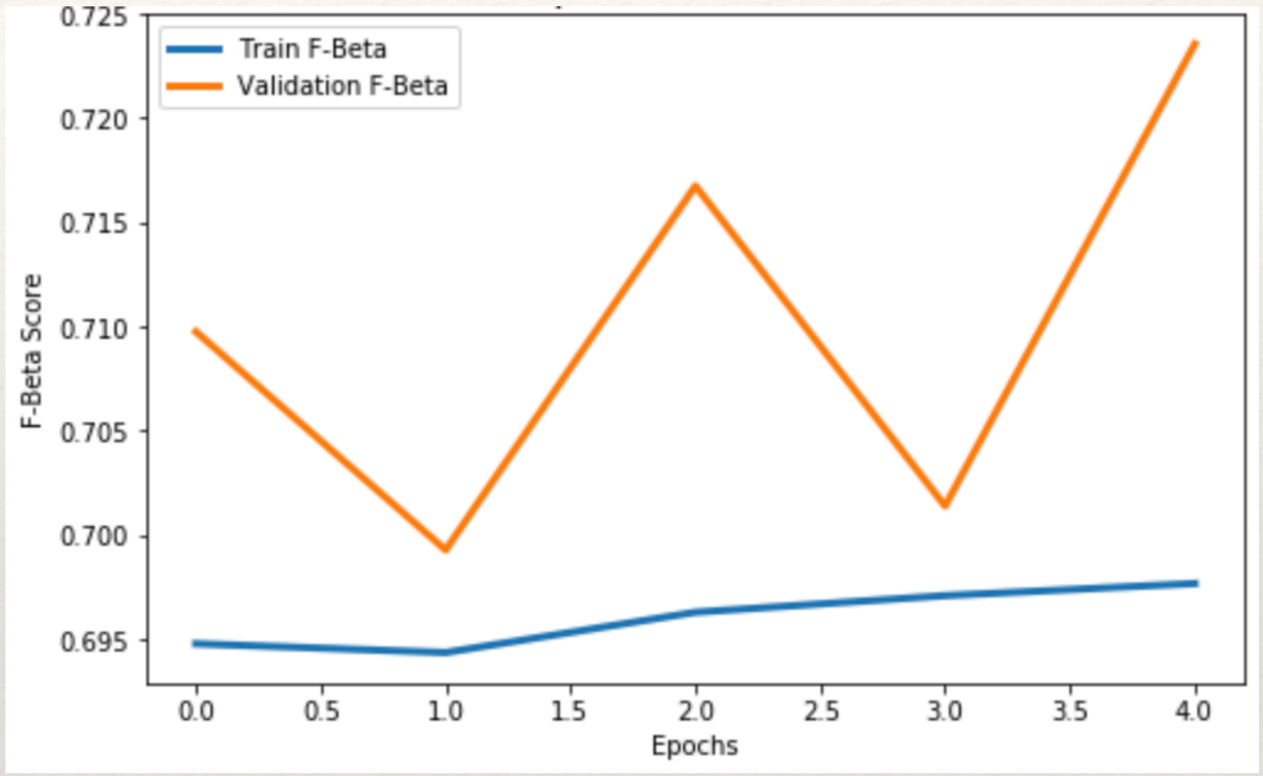
- Model performance continues to increase through successive Epochs.
- F-Beta score follows similar trends to accuracy, but is reduced and suffers from instability.
- Train score are more stable but lower than validation.

Model Results

Accuracy: Epochs 21 - 25



F-Beta Score: Epochs 21 - 25



- Upon additional Epochs (re-initialized with saved weights), we still see slight improvements in training performance but increased instability.
- This is evidence of "overfitting"

Final Accuracy: 0.938

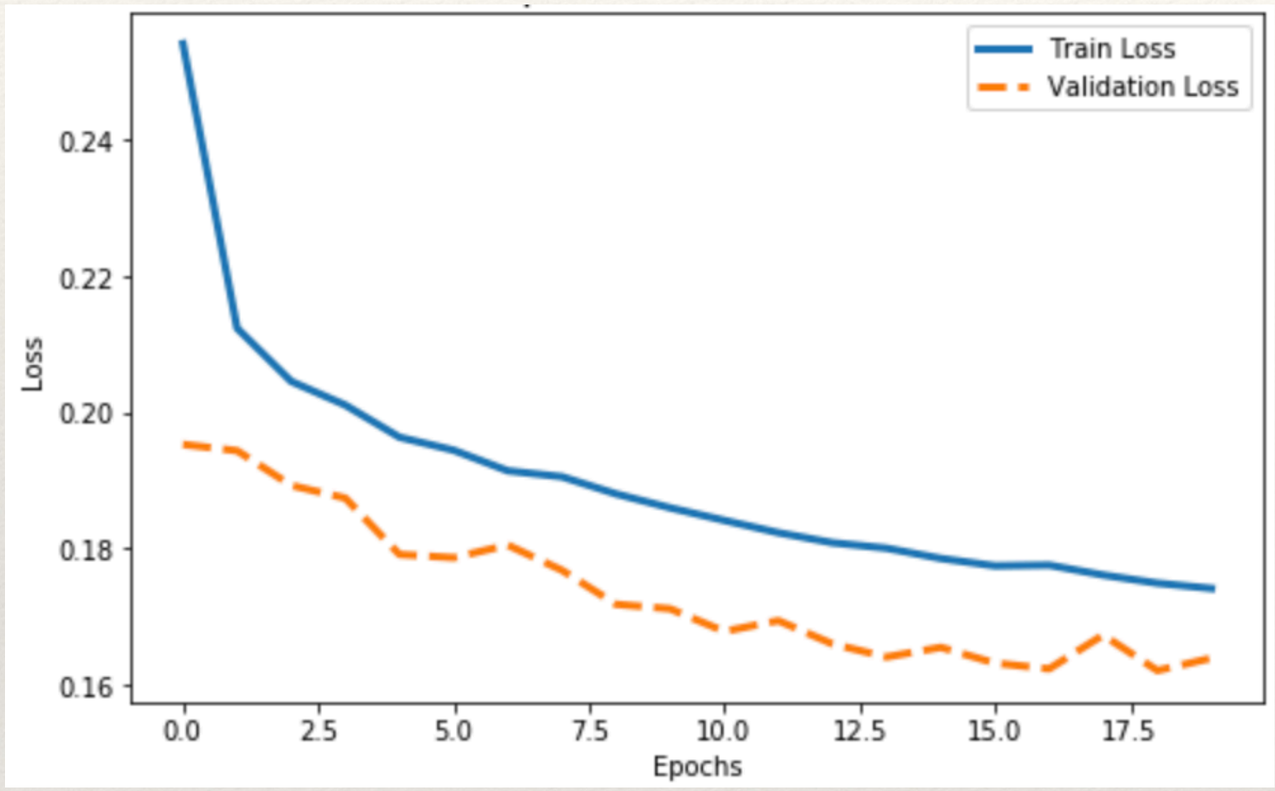
Final F-Beta:

0.846 Validation

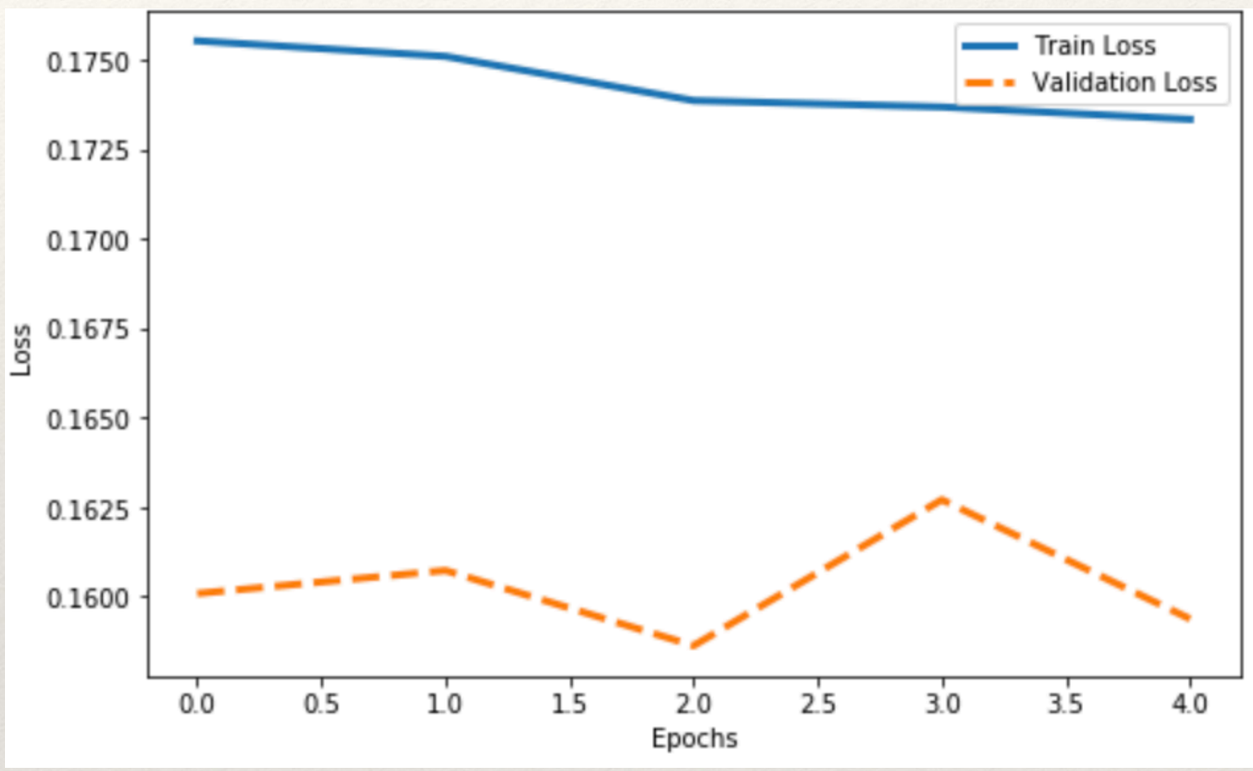
0.833 Test | 0.834 with Thresholds

Model Results

Loss: Epochs 1 - 20



Loss: Epochs 21 - 25



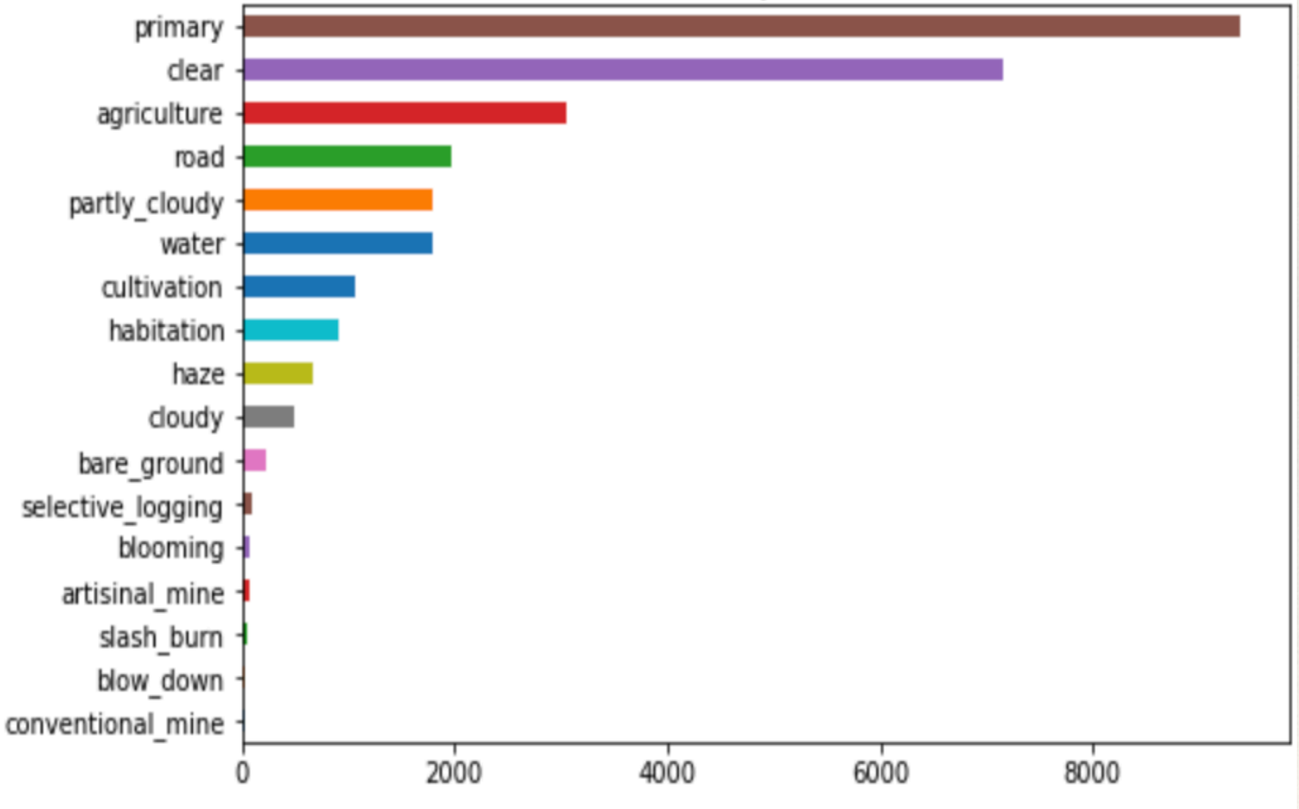
- Loss follows an inverse pattern to our other metrics.
- Shows general improvements with increasing epochs and instability after ~ epoch 22.

Validation Labels Investigation

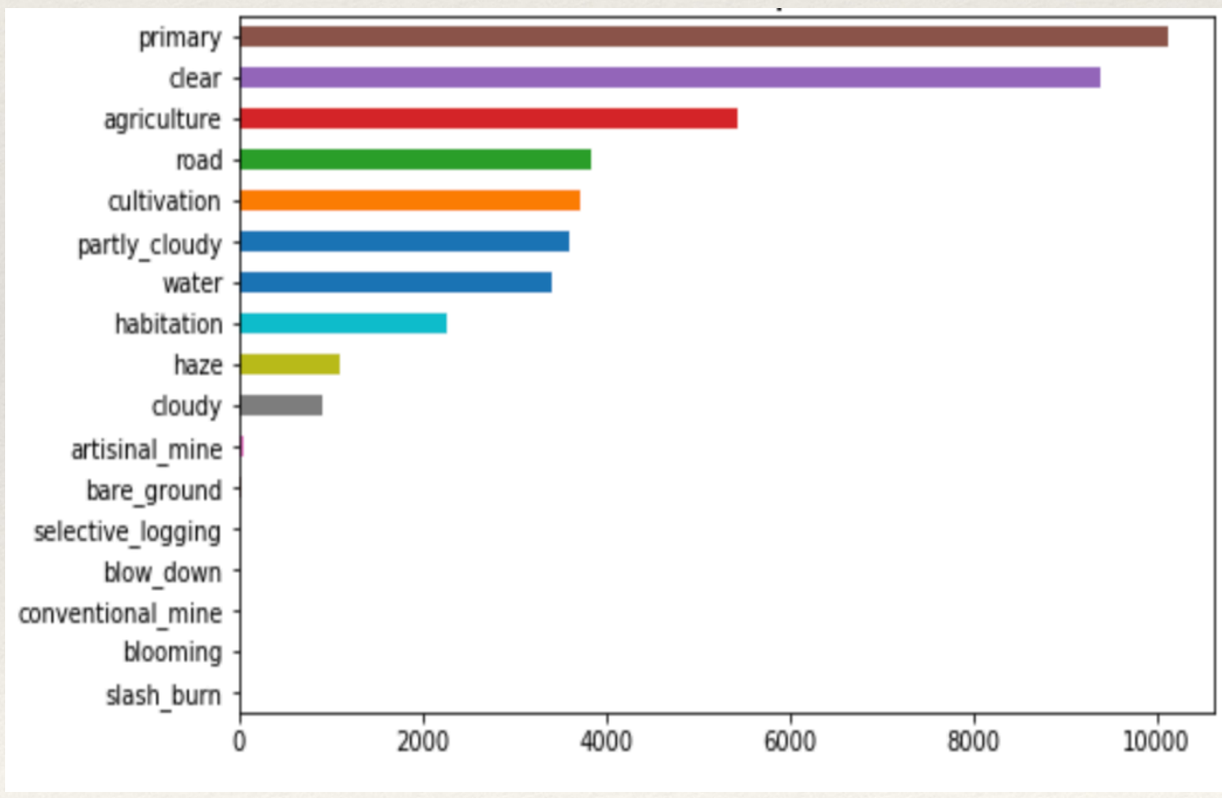
Validation Data: 10120 Images

- The model over-predicts labels
- This may account for the instability in F-Beta Score.

True Labels: 28902 Total



Predicted Labels: 43914 Total



Recommendations

- Invest in powerful GPU work-center.
- Continue training model with bootstrapped images (varying augmentations of low frequency label combinations).
- Increasing Network complexity.
- Consider practicality of performance - resource tradeoff.
  - Better performing models = more time & money.
- Partner with scientists for future conservation applications.

Acknowledgments

- Matthew Brems: seamless introduction to CNNs
- Riley Dallas: introduction to batch processing.
- David Yerrington: troubleshooting linux, server issues and model.
- Tucker Allen: endless guidance and project scope adjustment suggestions to keep my vision within the bounds of planet earth.
- Isa Cuervo: presentation suggestions.
- All of my classmates for being awesome and helping me learn SO MUCH during the last 12 weeks!

Questions & Discussion

THANK YOU !

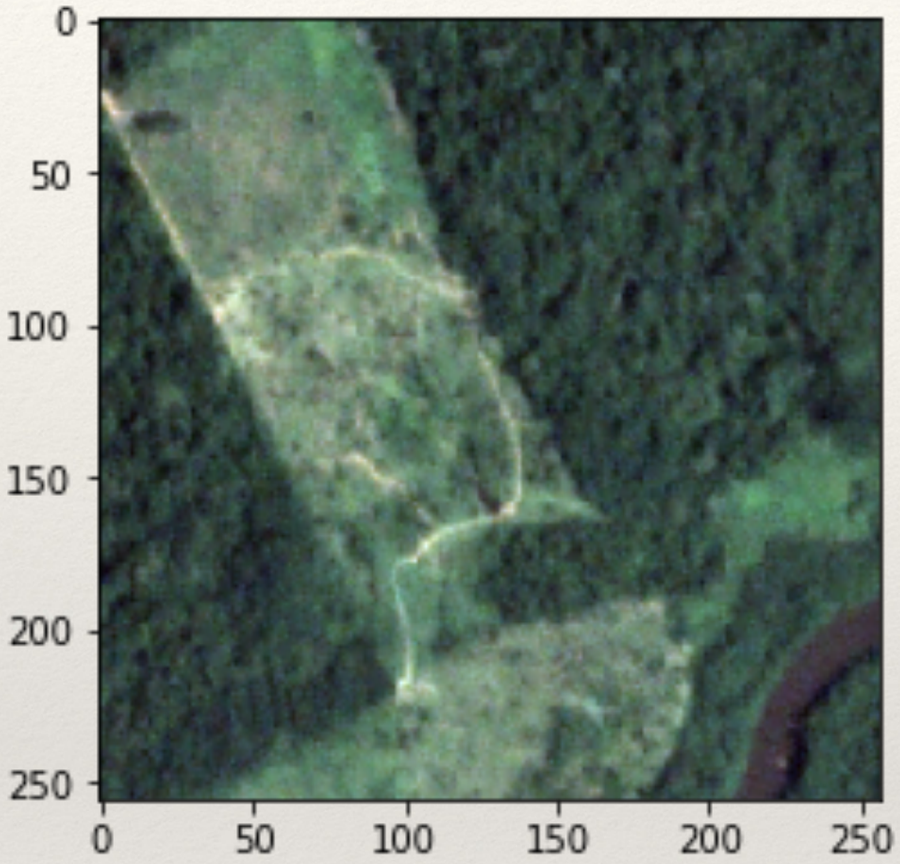
Model Structure

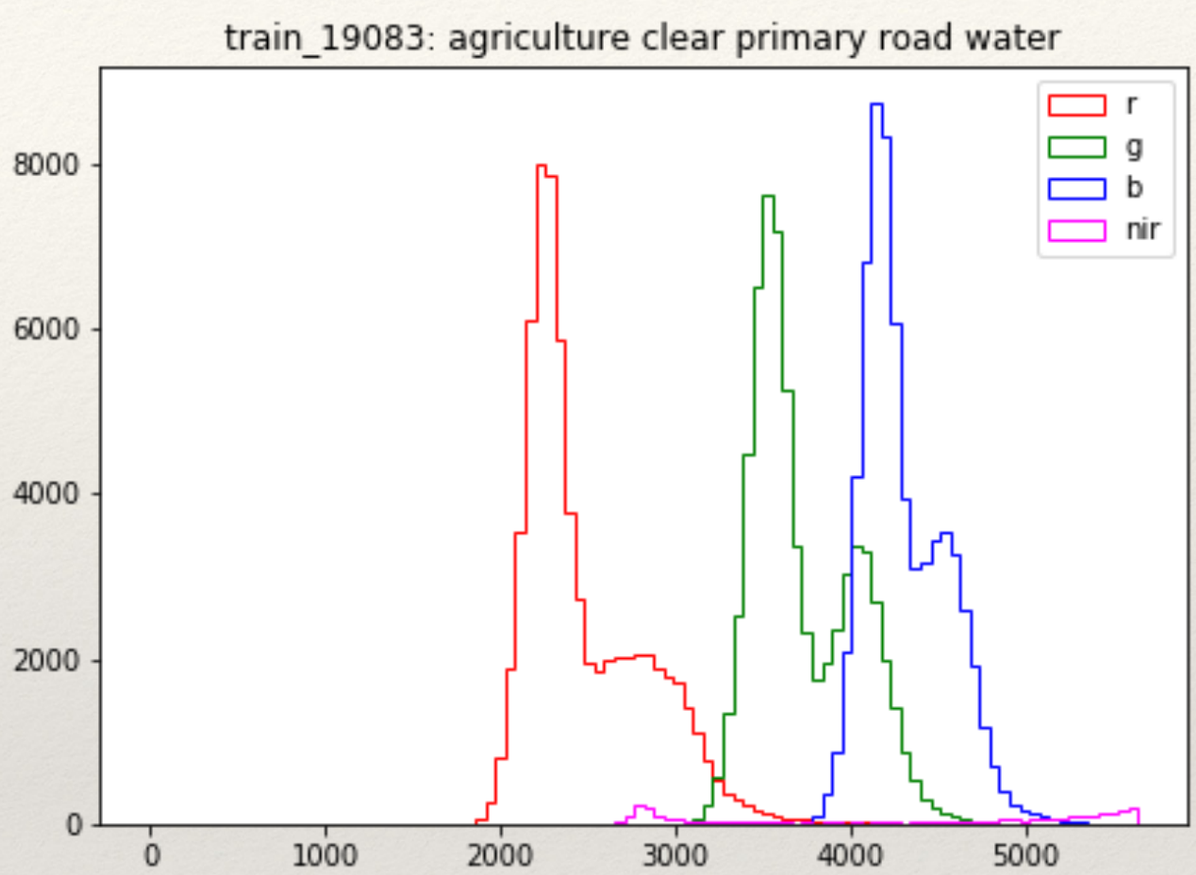
- C32 - Pool - C64 - Pool - Dense 128 - Dense 17
- Batch Normalization scales data to a .
- Dropout eliminates a random 50% of the nodes at each layer every epoch.
    - This reduces tendency for the neural network to overfit on the training data.

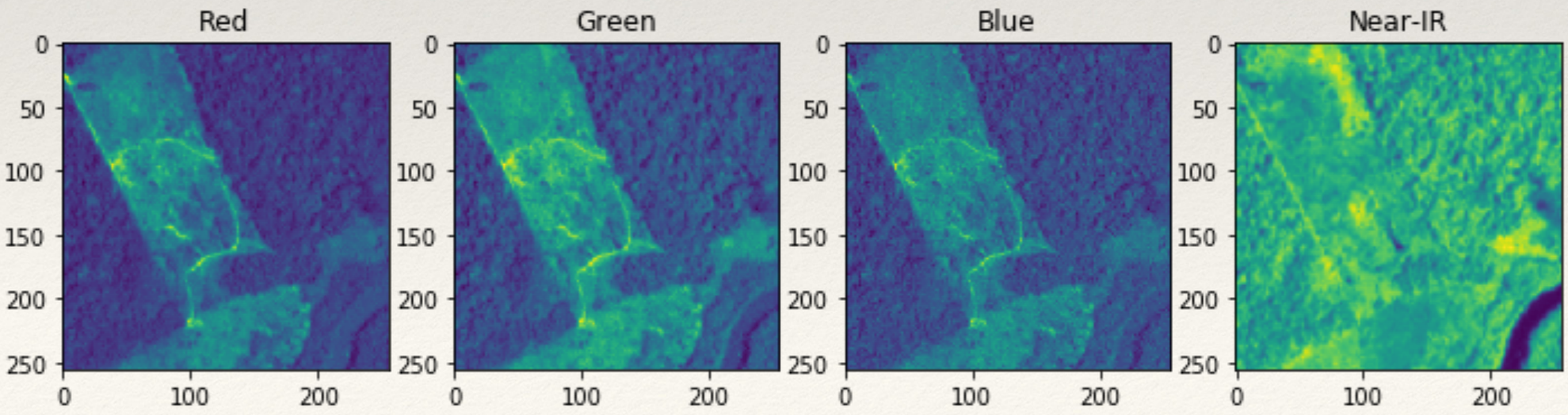
Batching

- Test data: 21 GB + Training data: 30 BG = OOM
- Need to do batch processing through and through, keeping consistent randomization across Kfold testing
- Thankfully, we have generators! Keras has a nice one to do all of this.

Understanding Image Data







Modeling Process cont.

- Regularization techniques
- Early stopping
- Dropout
- Batch normalization at multiple steps