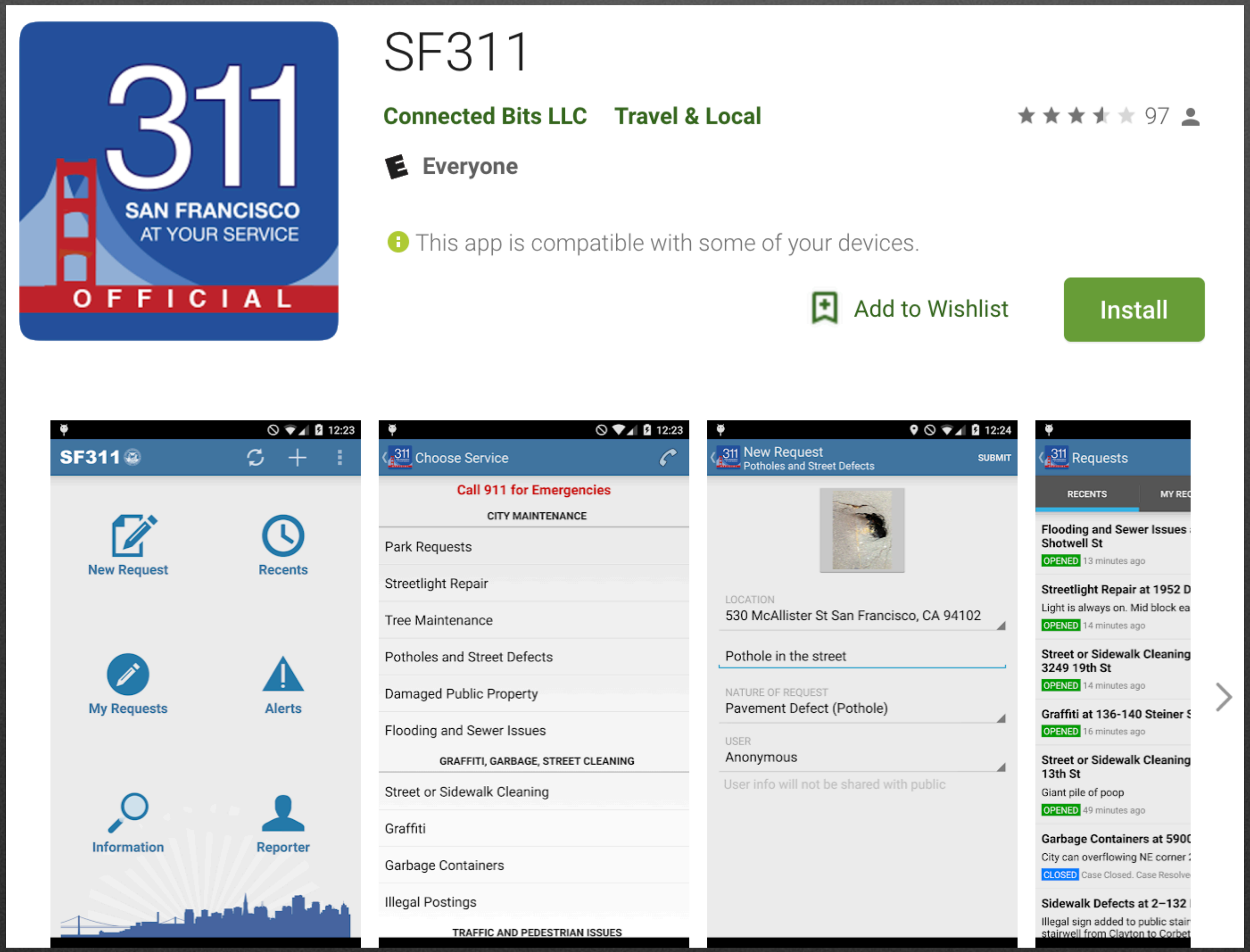
San Francisco 311 Reports Neural Network Image Classification

Crystal Bevis

San Francisco 311Cases

- 311 Reports - Non-emergency reports requesting city service
  - Example: Trash, graffiti, damaged roads
- Primarily created by mobile app
- Open data - images and categorical data about cases available



Objective

- Objective:
  - Determine the type of case by using the images, location, and date data.
- Motivation:
  - Make it easier to submit cases - take a picture and form can autofill
  - Audit cases - can predict which forms may need human review
  - Accessibility - all languages, vision impaired
  - Better allocate city resources

311 - Data

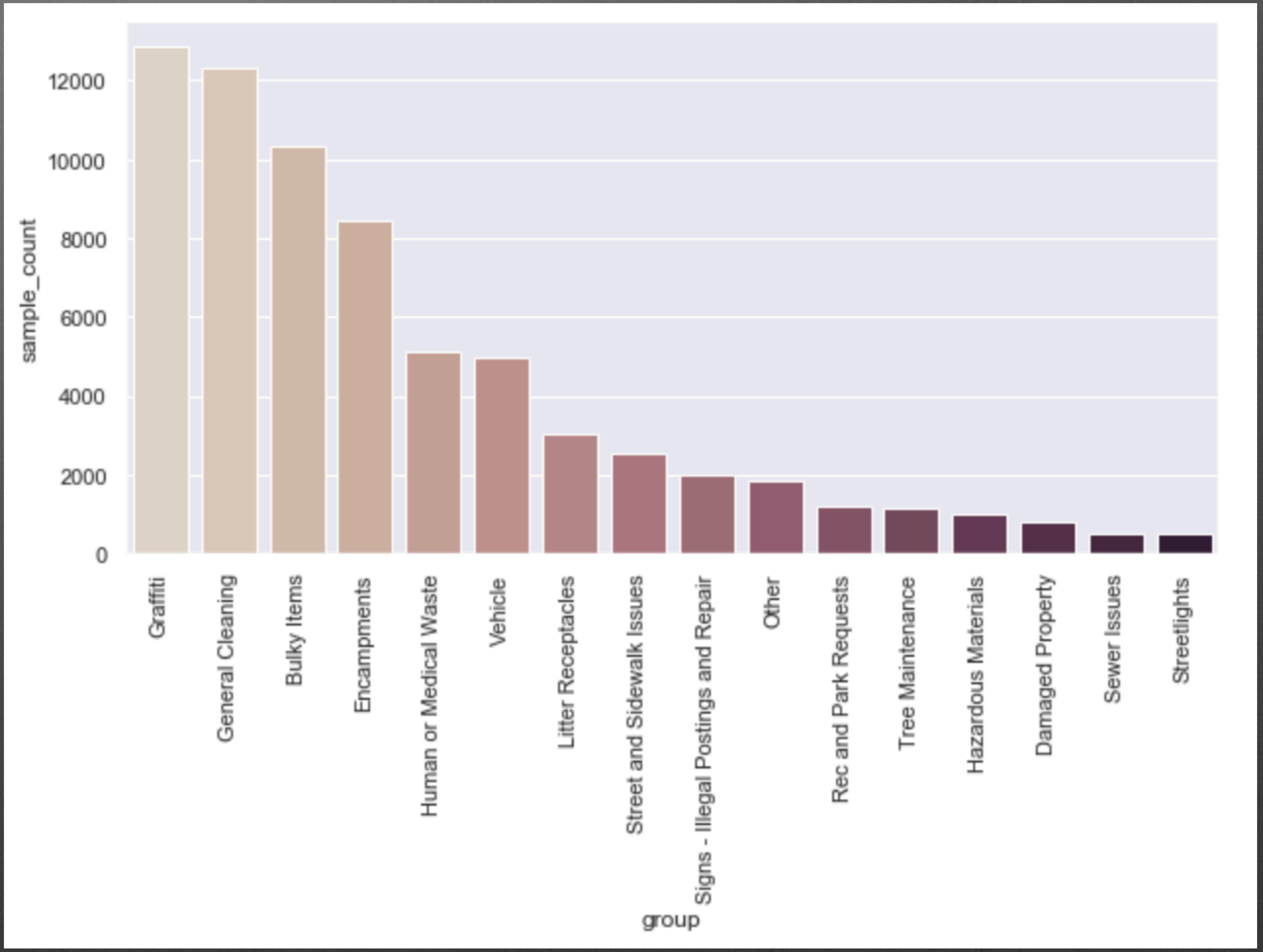
SF Open Data: https://data.sfgov.org/City-Infrastructure/311-Cases/vw6y-z8j6

- Large dataset - 3.48M samples, 20 features
  - Selected a subset to analyze due to memory and time constraints
- Images ~ 160,000 from 2017 and 2018
  - Cropped edges, resize, grayscale, and save as 128x128 images
  - Used PIL
- Categorical data selected
  - Street, Neighborhood, Date
    - Street and neighborhood could be generated from phone GPS
    - Date from phone as well

What kind of case?

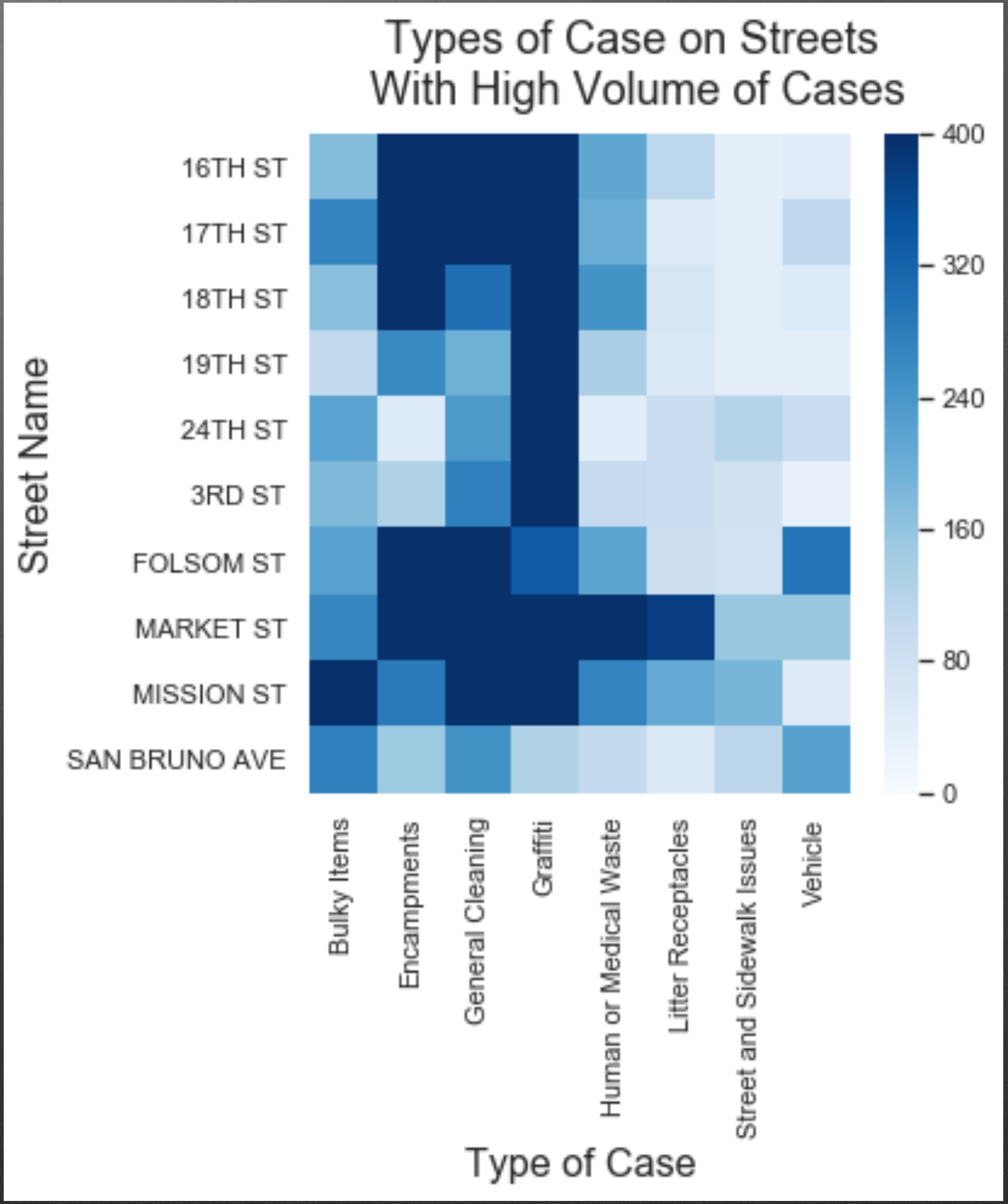
311 Data - Labels

- What to use as labels?
  - Category - 31 unique labels, ~50% in one class
  - Request Type - 119 unique labels
- Combined smaller categories, into groups
- Split largest category into request types
- Combined some obvious groupings
  - Ex: Parking enforcement, abandoned vehicles -> vehicle issues



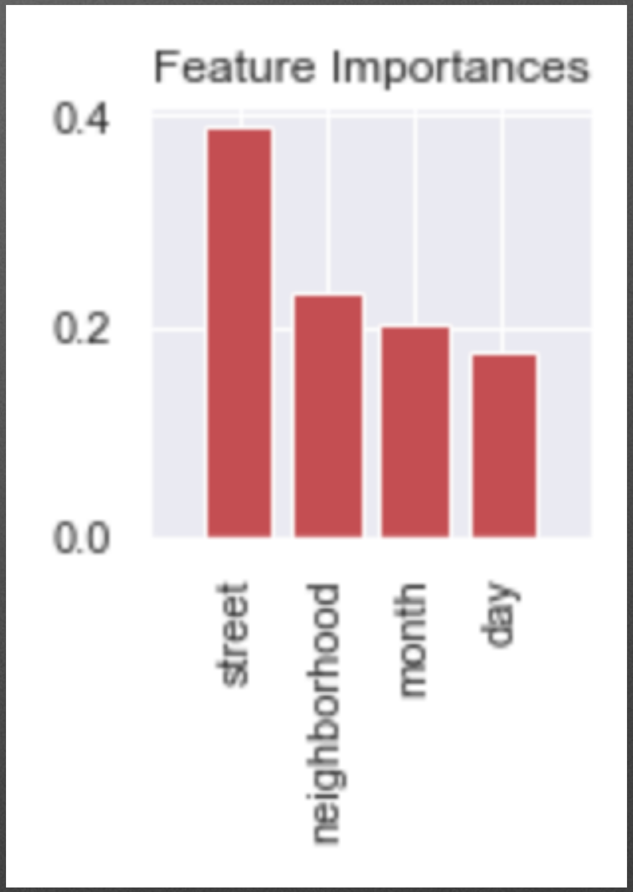
Exploratory Data Analysis

- Looked at categorical variables
  - Street, neighborhood, date
- Split date into month and day of the week
- Scale on heat map shown set to maximum of 400 to better see the distribution



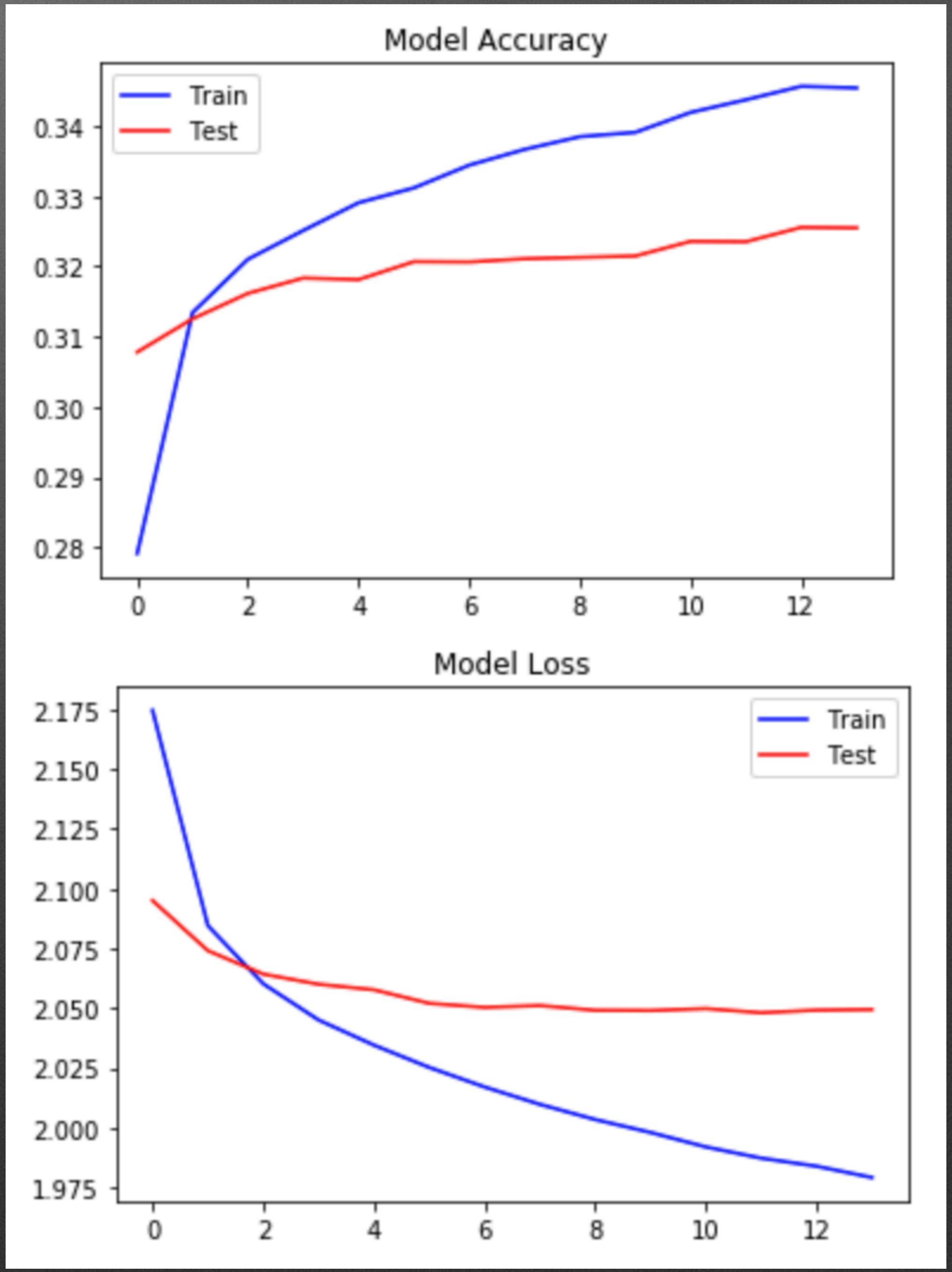
Modeling: Categorical Data Random Forest

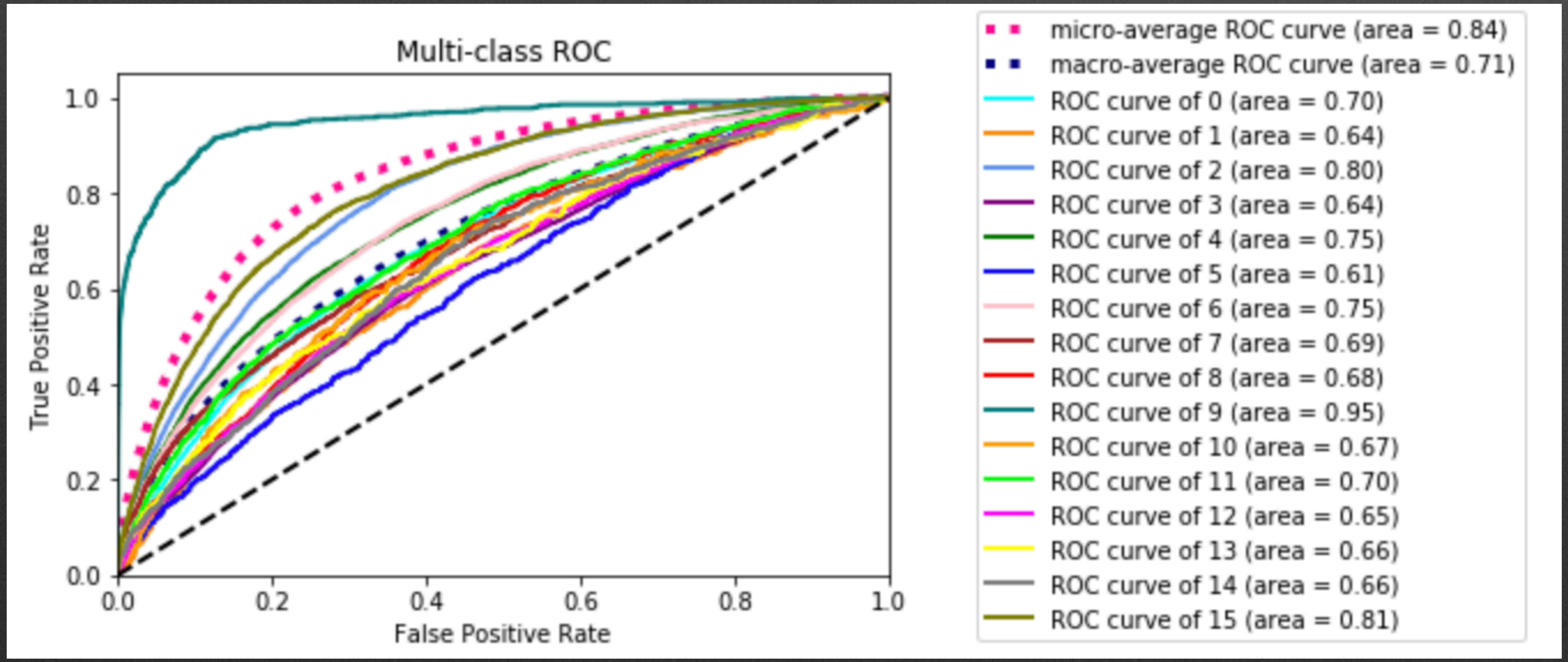
- Random Forests are good for looking at feature importances
- Handle categorical data
- Determine a baseline for
- Train accuracy: 38%
- Test accuracy: 32%
-



Multilayer Perceptron

- Top AUC:
  - 9: Parks and Recreation (0.95)
  - 15: Vehicles (0.81)
  - 2: Encampments (0.80)
- Bottom AUC:
  - 5: Hazardous Materials (0.61)



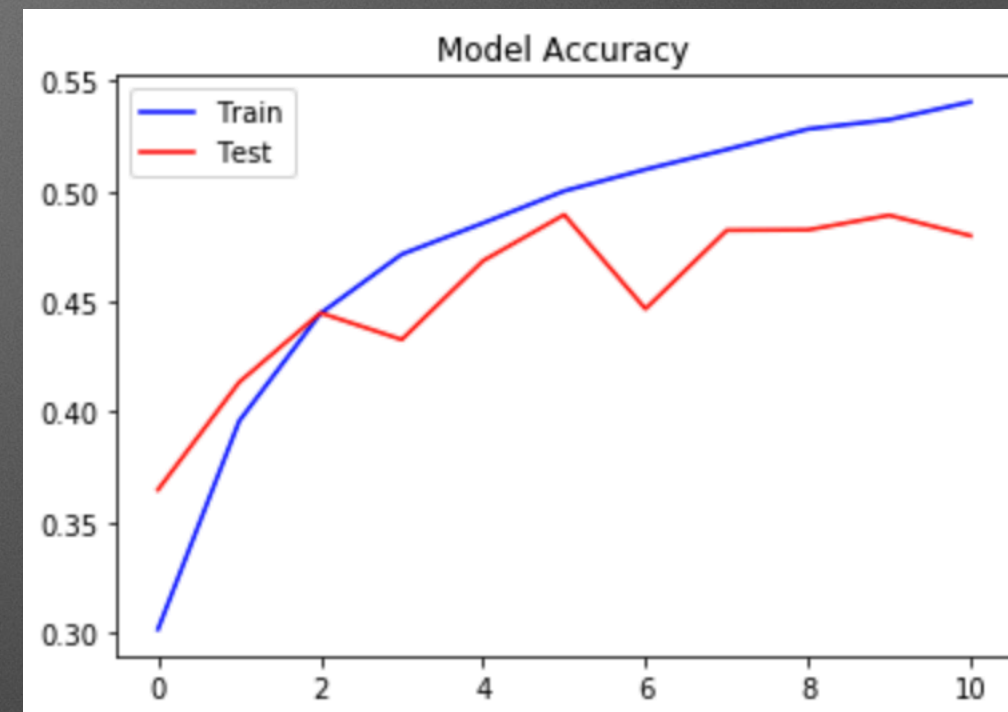


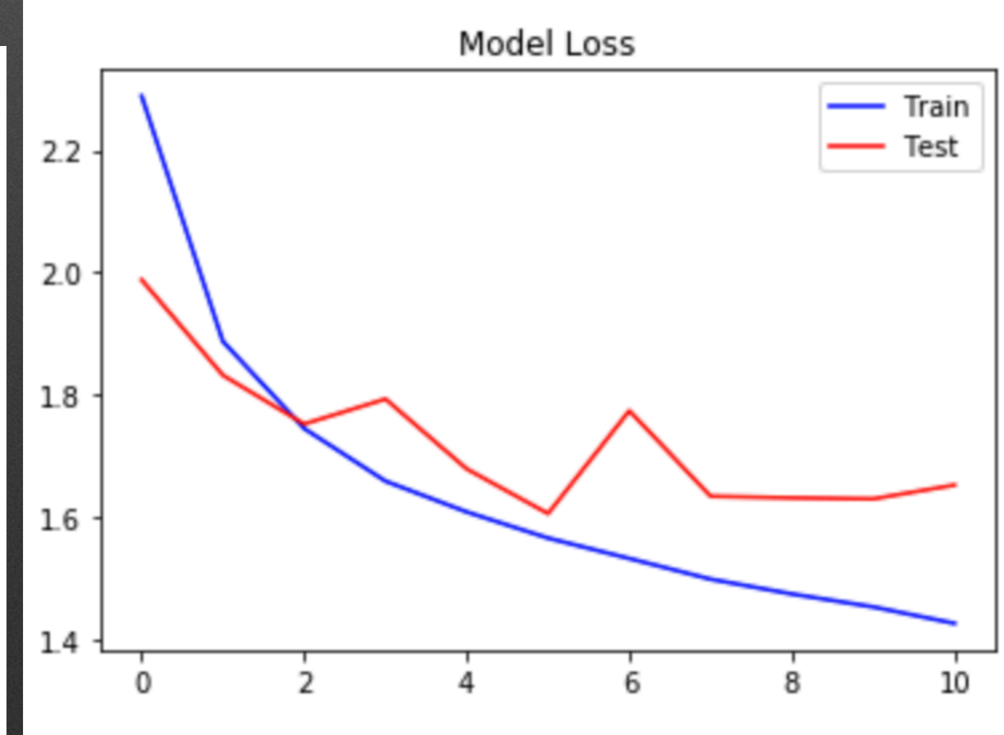
Convolutional Neural Network

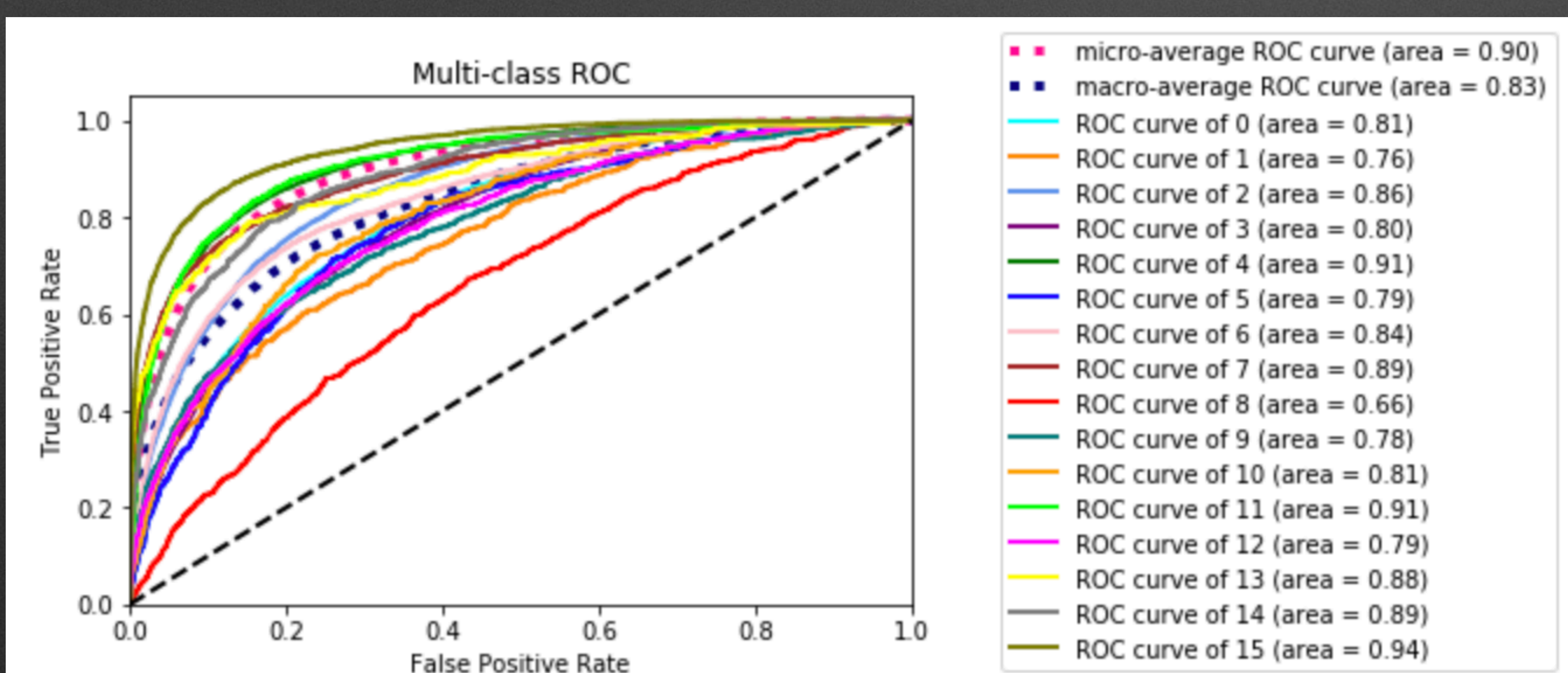
- Moved from local machine to Google Colab (14 hrs -> 15 min)
- 6 Convolutional 2D layers
  - Filter sizes [32, 32, 64, 64, 128, 128]
  - Kernel sizes (3 x 3)
  - ReLu activation
- Model overfits - used several techniques to reduce this
  - Batch normalization
  - Drop out
  - Max Pooling
- Early stopping, patience = 5
  - Stops after 11 epochs
- Training Accuracy: 0.54
- Test Accuracy: 0.48

Convolutional Neural Network Results

- Top 3 AUC:
  - 15: Vehicles (0.94)
  - 11: Signs - Illegal Postings and Repairs (0.91)
  - 4: Graffiti (0.91)
- Bottom AUC:
  - 8: Other (0.66)



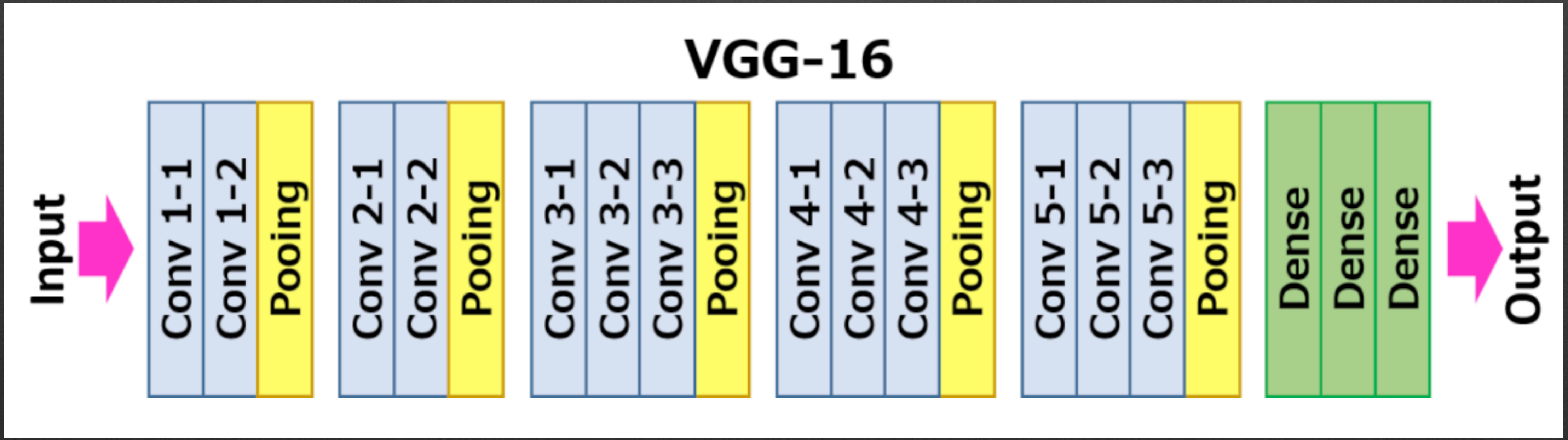




VGG 16 - Images

- Very Deep Convolutional Networks for Large-Scale Image Classification
  - Top 5 ImageNet classifiers
  - Easy to implement
  - Over 533MB
- Only accepts RGB inputs - triplicated my grayscale data
- Early Stopping

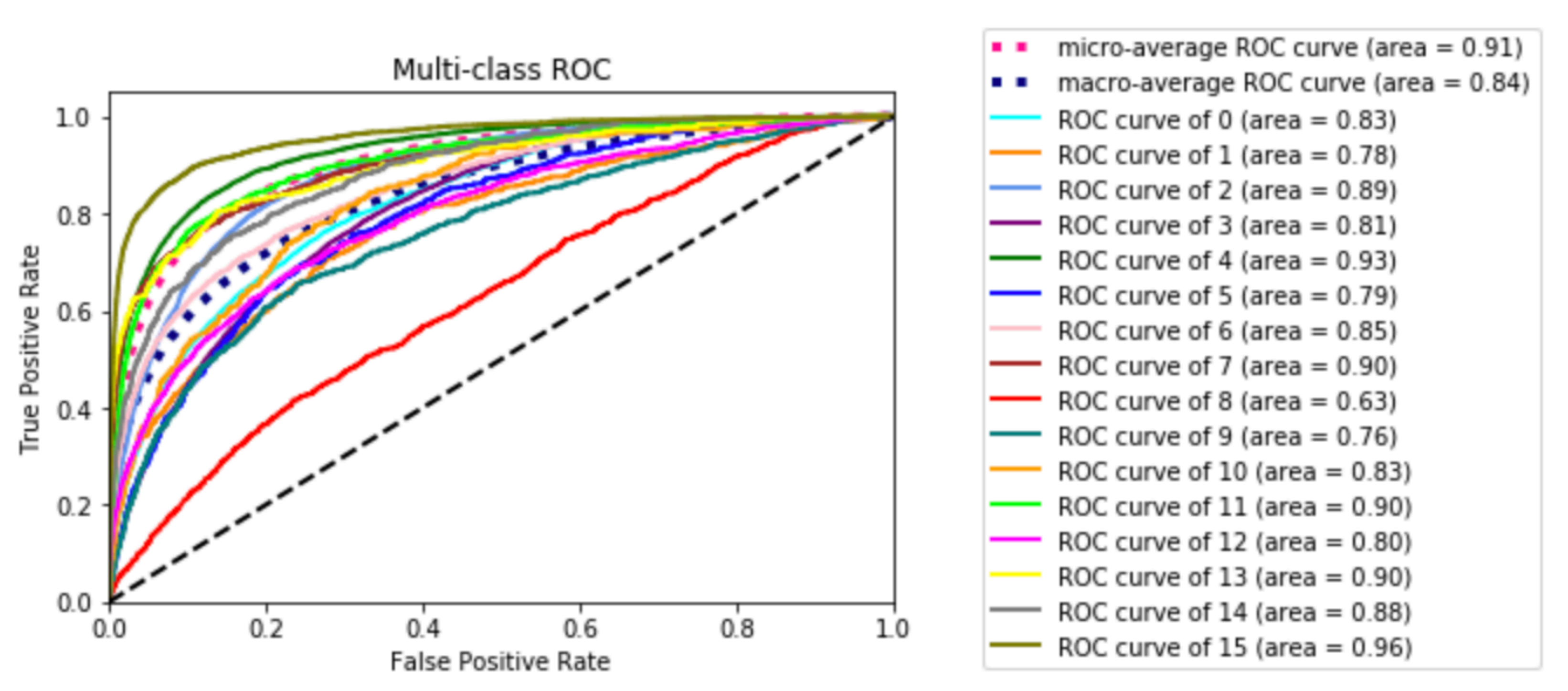
- Training Accuracy: 0.78
- Testing Accuracy: 0.53

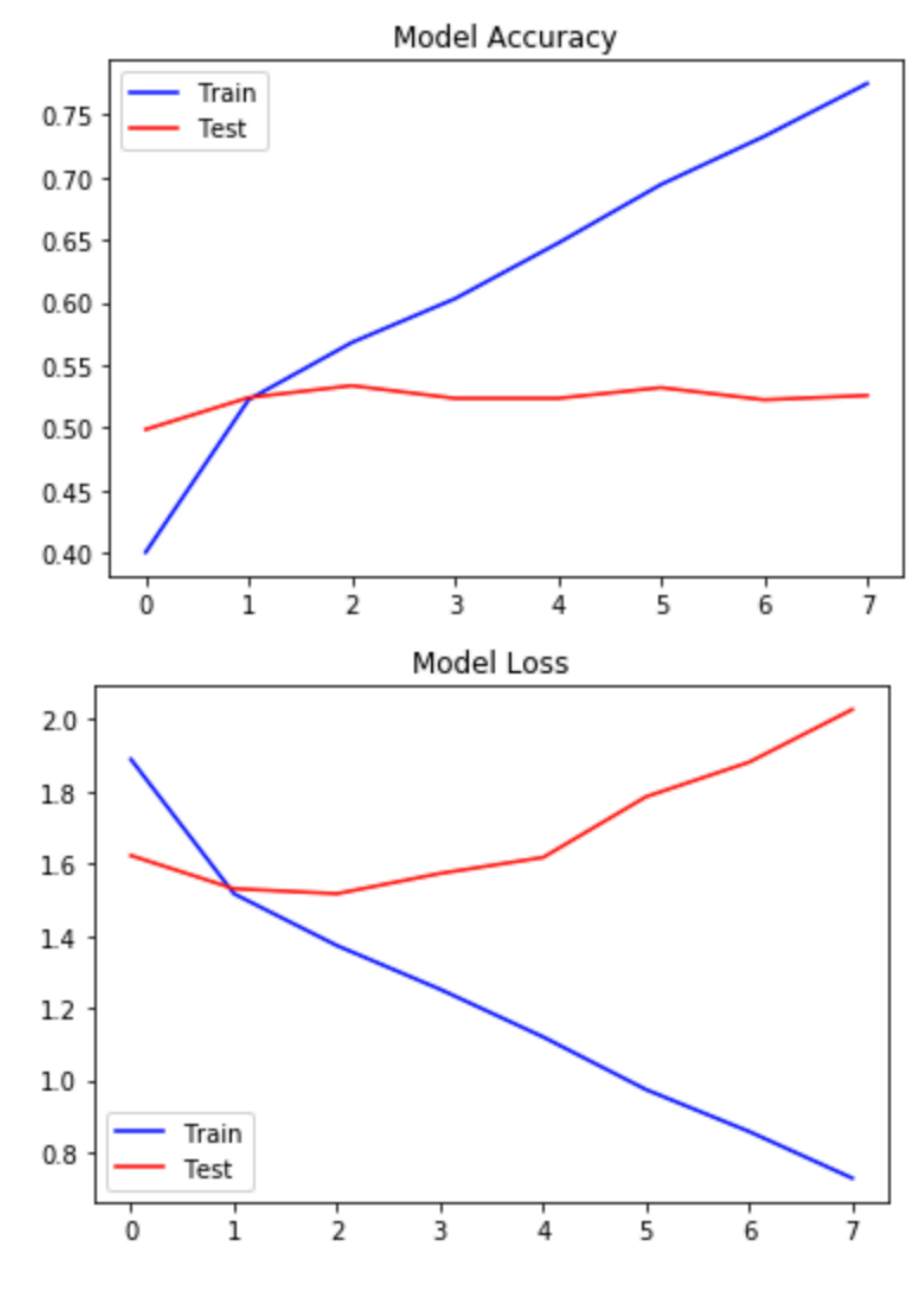


https://neurohive.io/en/popular-networks/vgg16/

VGG16 Results

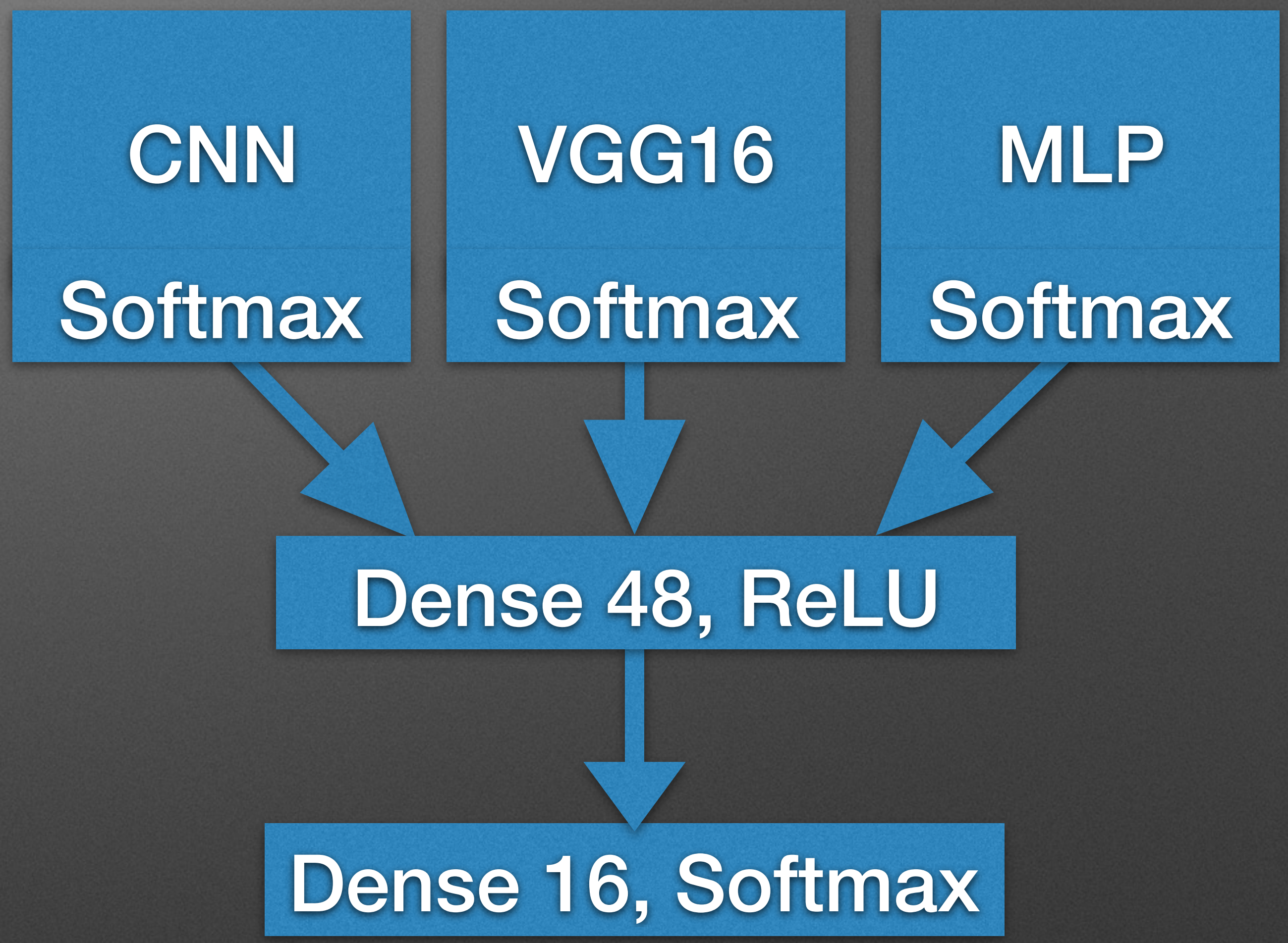
- Top 3 AUC:
  - 15: Vehicles (0.96)
  - 4: Graffiti (0.93)
  - 7: Litter Receptacles, 11: Signs - Illegal Postings and Repairs, 13: Streetlights (0.90)
- Bottom AUC:
  - 8: Other (0.63)



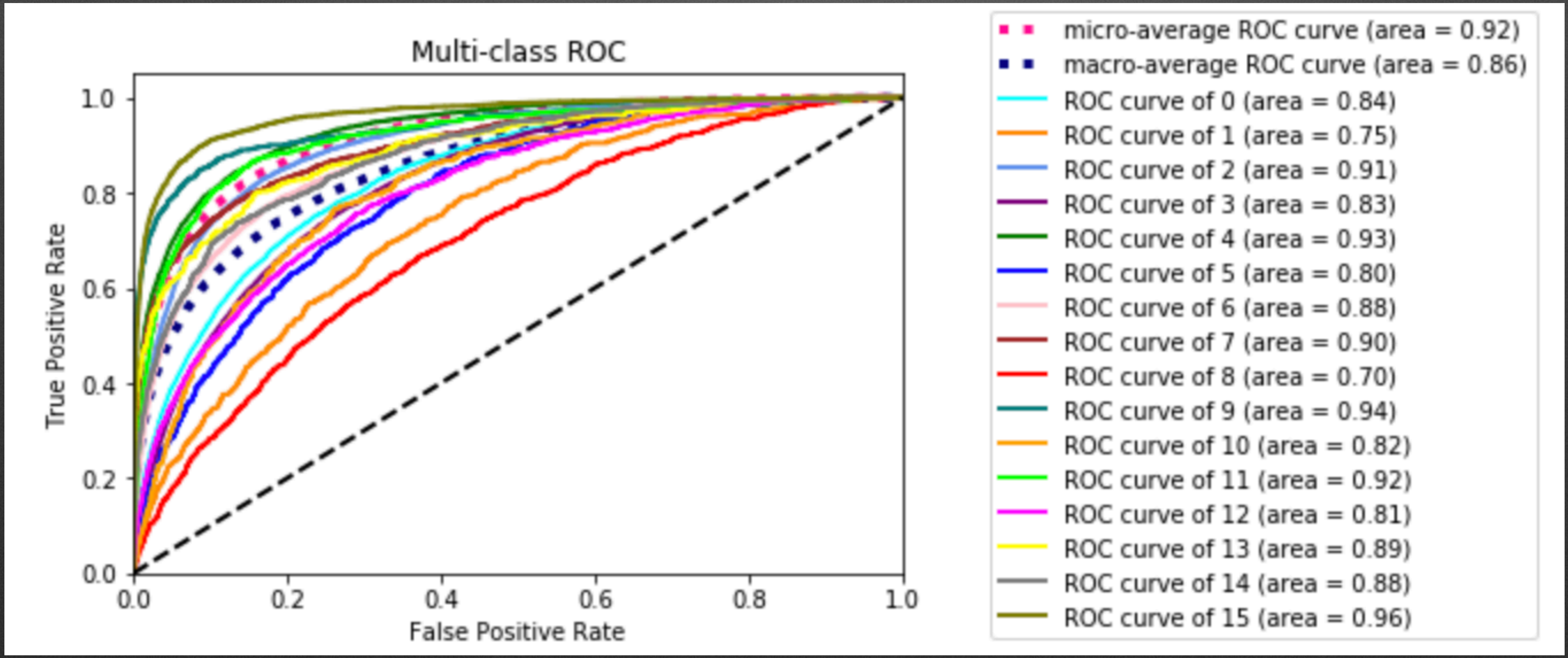


Multiple Data Types Combine Model

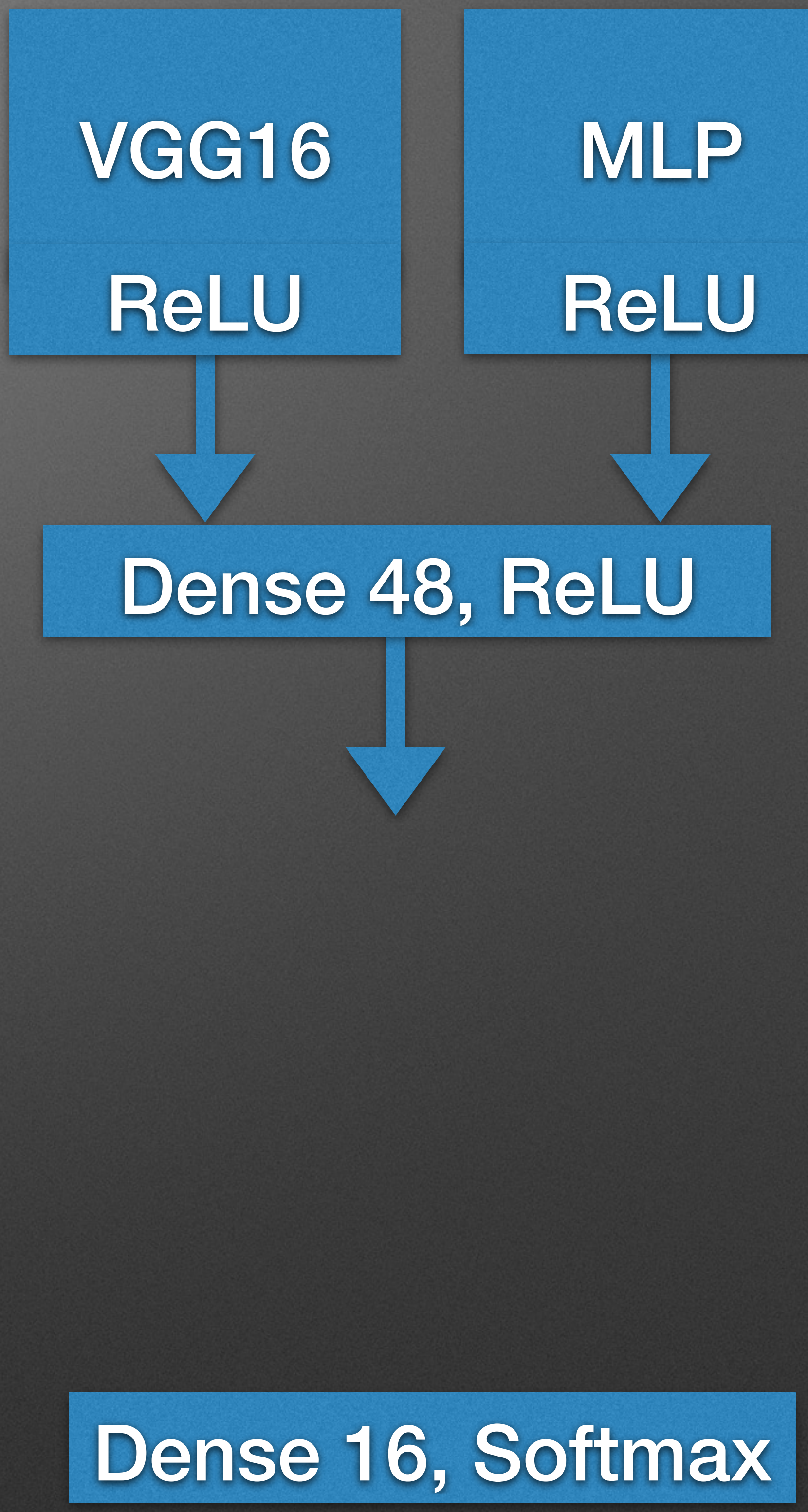
- Train accuracy: 62%
- Test accuracy: 54%



Multiple Data Types Combined Model



Multiple Data Types Combine Model



- Changed outputs of individual models to ReLU
  - Softmax not needed until the end

- Changed
- Train Accuracy: 0.54
- Test Accuracy: 0.54

Future Work

- Expand to include time series information across years: 2013-2019
- Include more images
- Try with RGB images
-

Thank you!