

# Sketch Classification

...

Andrew Bergman  
4 November 2019

# Agenda

- ❖ Game Background
- ❖ Problem Statement
- ❖ Preprocessing
- ❖ Modeling
- ❖ Evaluation
- ❖ Conclusions

# Game Background

- ❖ Quick, Draw! is a game developed by Google to help with machine learning. Players are prompted to draw 6 items & they are given 20 seconds to draw each item
- ❖ Google stores the drawings & makes them open source on their Creative Lab's [GitHub](#)



# Problem Statement

- ❖ Image recognition has a wide variety of uses from self-driving cars to interpreting X-rays. Classifying handwritten characters is one of the most common uses & is used everyday in places like the Post Office. However, it is difficult to do correctly because nobody writes, or in our case draws, the same way & misclassification of a character can be significant.

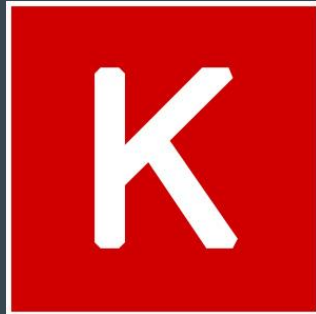
# Preprocessing

- ❖ Google offers the Quick, Draw! data in various formats
- ❖ Their raw files contain metadata we used in EDA
- ❖ We decided to use the Numpy bitmap format for modeling
  - The images were resized & centered
- ❖ Our preprocessing was minimal: we just had to scale & resize the data
- ❖ The data is stored in an S3 bucket for the modeling stage



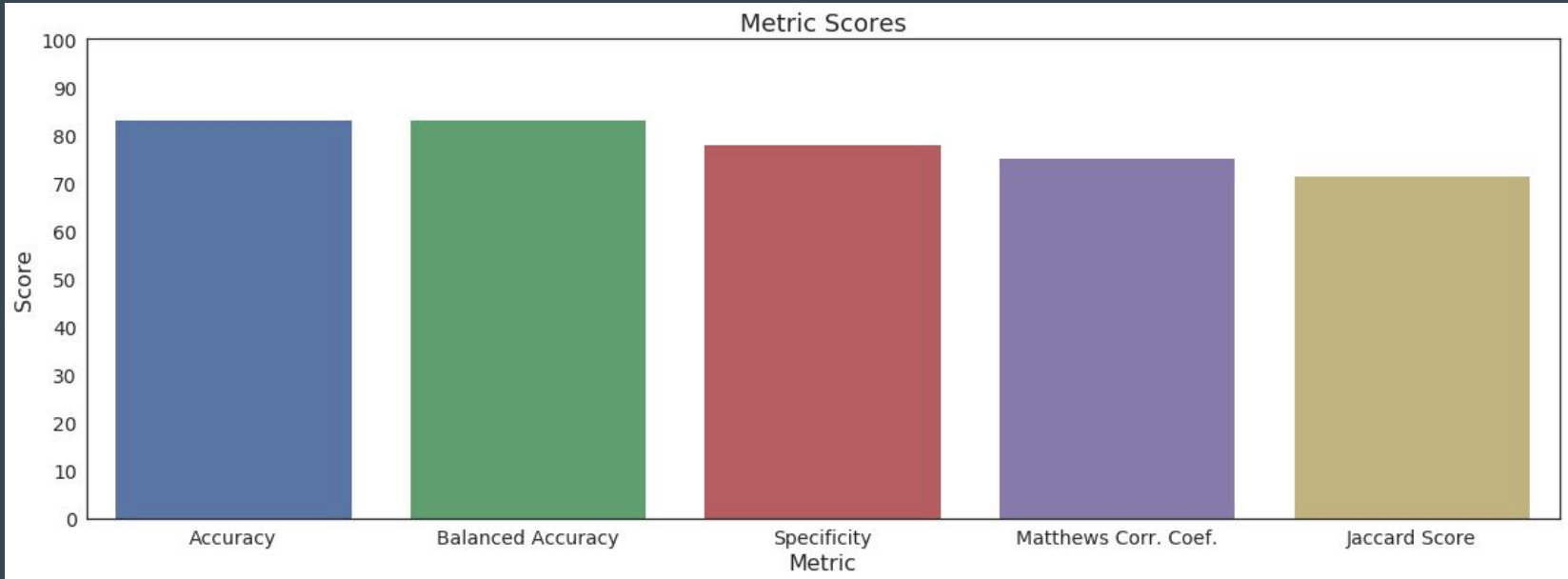
# Modeling

- ❖ Convolutional portion: 2x convolutional layers + 1 MaxPool layer
- ❖ Dense portion: 2 dense layers
- ❖ We made heavy use of regularization:
  - $L_2$  was included at each node
  - Dropout & batch normalizing layers were added after each dense layer
  - Early stopping was included when we fit the model
- ❖ The model was run in an AWS SageMaker instance



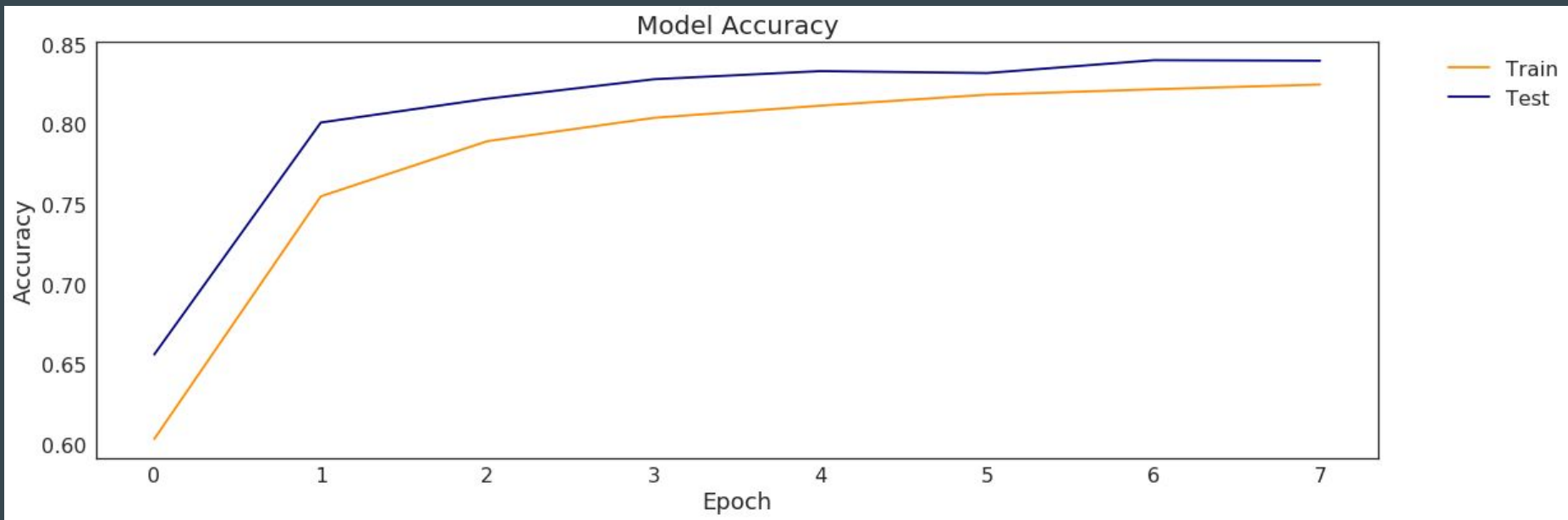
# Evaluation

- ❖ The model was evaluated in three ways: metric scores, loss, & accuracy
- ❖ We could not create an ROC curve because we have 3 classes



# Evaluation

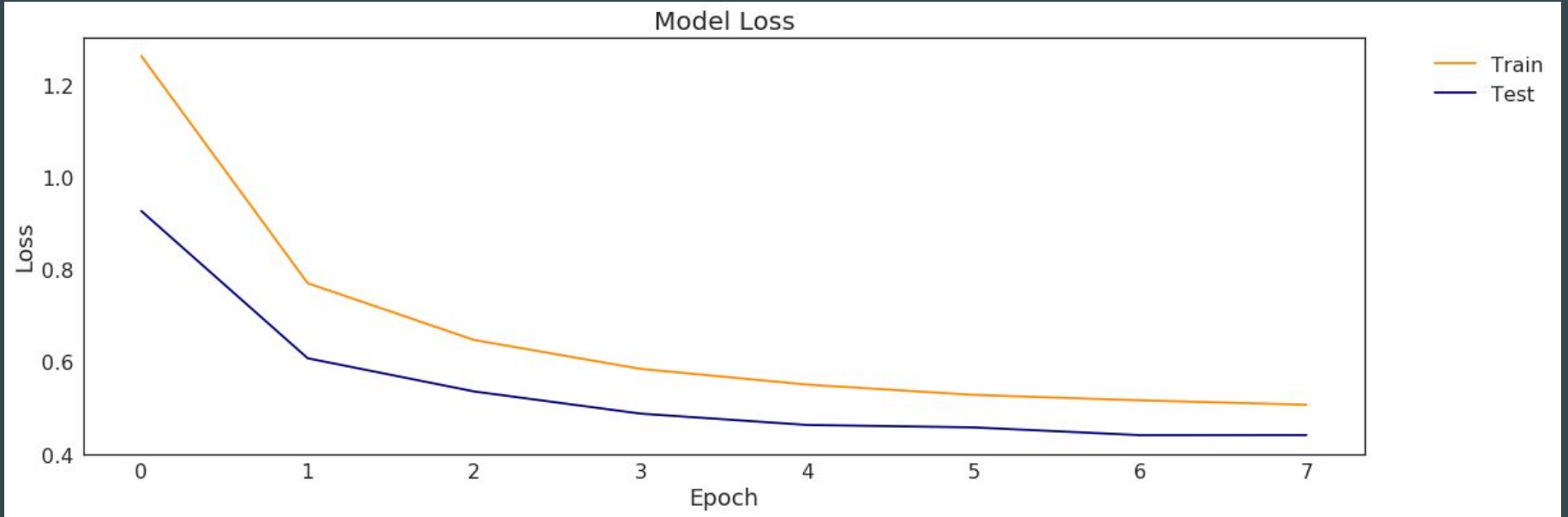
- ❖ Sharp improvements but stayed relatively flat





# Evaluation

- ❖ Sharp decrease in loss into the third epoch which then flattened but then increased going into the eighth when the early stopping took effect.



# Conclusions

- ❖ Our model had okay performance, but has room for growth
- ❖ The data is not ideal: Google does not say *how* to draw the animals
- ❖ If the data were more consistent we would have better performance
- ❖ The images were also small: only 28x28 pixels
- ❖ Classifying the images is possible, but the data have to be better