Using Deep Learning to Classify Images of Paintings

Juliana McCausland

The goal of this project was to create a model that is able to identify the artist behind various famous artworks. This project is a first step in the process of eventually learning to train models to authenticate artwork and identify forgery. I created a CNN model using keras to classify a dataset of Impressionist paintings into ten different categories. The final model ran with about 67% accuracy, and a validation loss of about 1.0. The primary takeaway is that Impressionist artists have similar works, and there are many difficulties when attempting to differentiate these paintings.

Data

I used a dataset from Kaggle that contains Impressionist paintings by ten different famous artists. There are about 400 images for each artist (class). The images were already separated into folders for each class, and there was a training set and validation set. The images were mostly colored, and all of them were similar stylistically.

Algorithms

I first created a baseline CNN that was built directly on top of the ResNet50 pretrained model. This model had an accuracy of 55%, and a validation loss of 3.5. The model also seemed to be overfitting. I modified the model by using the he_uniform kernel initializer, adding batch normalization, and increasing the number of units. The accuracy immediately increased to 72% and the loss decreased to 0.9, but there was a large gap between the train and validation accuracy scores. I continued to tune the hyperparameters to address the overfitting (as well as using reduced learning rate) and to increase the accuracy. At one point, I had a model with a maximum of 500 units and dropouts (of 0.3) that resulted in a 73% accuracy score with 0.9% loss. This model seemed to be heavily biased toward one artist during every test. Each time I trained the model, it would choose another artist for its bias, and would thus output that one artist for each test image. I attempted to change loss functions, optimizers (I tried SGD as it had successfully circumvented this particular issue for others), and the learning rate (0.001). I eventually landed on a model that had less of a bias problem, but that still struggled to distinguish between artists. This final model had an accuracy of about 67% and a validation loss of about 1.02. My own tests on the model using outside images did not reflect this accuracy score. Most of my tests were incorrect. I created a classification report to diagnose the bias problem and the low accuracy (while testing, despite the validation accuracy of 67%) and saw that the but this report indicated very high precision, recall, and f1 scores for every class, with balanced class supports. I am stumped on this.

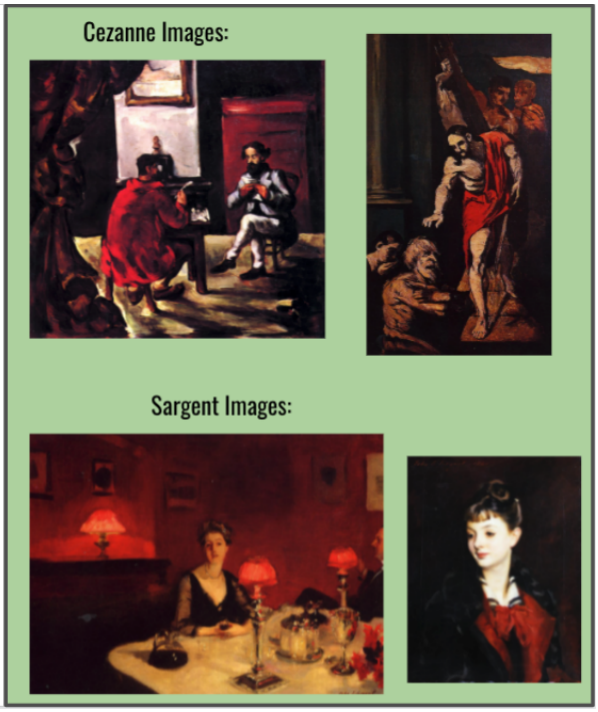
Tools

I used tensorflow, keras, jupyter notebook, and numpy

Communication

Below are screenshots of slides highlighting incorrect classifications, with photos from the training sets of the incorrectly identified artist, and the correct (but unidentified by the model) artist.

Artist: Cezanne (probability: 10.14%)



Artist: Matisse (probability: 29.07%)