Mariam Joan
August 10, 2019
Udacity Data Analyst Nanodegree
Project: Wrangling and Analyze Data

In the final part of this project we were to store and analyze our tweet data. For storing, I utilized sqlalchemy's create engine which allows you to store a sqlite database locally and convert your Pandas DataFrame to SQL which you can then query and have stored safely.

For analysis, I made sure to include Pandas libraries like matplotlib and seaborn. In the first visualization, we are viewing retweet count frequency which looks like over 750 retweets is common and similarly with the next visualization.

Then using seaborn, we could plot both retweet count and favorite count to see similarity which we see in the x, y axis with some outliers.

The next set of visualizations were to focus on how many times the neural network prediction outcome was true versus false. As you can see with each bar chart image the prediction outcomes were mostly true for first, second and third prediction outcomes. The very first bar chart uses extra logic to attach percentages with each bar so we can see in the first prediction outcome, 73.5% was true versus 26% were false.

The archive data did show that the neural network sometimes didn’t actually predict a dogs name. Sometimes you would see object names because a dog was in a car and so the prediction name was then a car. I wanted to gauge how many times the neural network appropriately guessed an actual dog breed. I found and cited a Github repo that had a basic text file of dog breed names which I converted to a list. From there, I combined all predicted names from our archive data and merged them to a unique set. Then we compared this unique set with actual dog breed names which produced 78 correct matches.

I also wanted to see tweets likes over time, and so I made sure to convert the date column into a datetime data type, and then create a new DataFrame with dates, and tweet likes. With seaborn we can see that the highest likes occurred during mid June 2016.

Lastly, being that we had the tweet text and counts I was able to find and cite code from a developer who wrote logic to find top retweet tweets and favorite liked tweets. Here we can read the tweets that were in the top 5 ranked by count. Using Python's collections library and the Counter class with most_common function we can also see from tweet text what the most common words used within our dataset.