Relatório

Nome: Leonardo Soares Duarte

Este relatório faz parte do Projeto 2 do Nanodregree Fundamentos de Data Science II – Udacity

1) Coletando

A princípio foi executada a etapa de coleta de dados, onde obtive os dados de 3 formas:

- A primeira foi através do download manual do arquivo 'twitter-archive-enhanced.csv' que contém a maior parte dos dados.
- A segunda forma foi baixando programaticamente o arquivo 'image-predictions.tsv' através da biblioteca Requests e salvando-o em um arquivo com o mesmo nome.
- Por último, foram usadas as bibliotecas Tweepy e JSON para autenticação de acesso e obtenção dos dados através da API do Twitter. Esta etapa demandou um pouco mais de esforço. Após a autenticação,a função 'get_status' foi usada para obter as informações de cada 'tweet_id' do dataframe 'tweets' e em seguida salvando-os em um arquivo de texto. O JSON de cada tweet foi salvo separados por uma quebra de linha (\n).

Após a obtenção, os dados foram carregados como dataframes do Pandas.

2) Acessando

Nesta etapa, os dados foram explorados em busca de problemas de qualidade e arrumação seguida da sua documentação. Começando pelo dataframe 'tweets'.

Quanto a qualidade dos dados:

- O primeiro problema detectado foi que havia numeradores e denominadores da nota (atribuída pelo autor dos tweets aos cãezinhos) que eram muito grandes ou muito pequenos;
- A coluna 'timestamp' estava como um texto ao invés de DATETIME;
- Analisando a coluna 'text', havia um link inativo que poderia ser retirado;
- Foram identificados 2 problemas na coluna 'name'. Havia nomes com o texto 'None' quando deveriam ser valores nulos
- Nomes inválidos, como 'a', 'an' e 'the', que sempre iniciavam com letra minúscula;
- Alguns cachorros possuíam mais de uma classificação de estágio (doggo, floofer, pupper e puppo);

Já na arrumação dos dados:

- Havia colunas que não seriam utilizadas;
- E também que a variável estágio estava distribuída em 4 colunas, quando deveria ser apenas uma.

Partindo para o dataframe 'image_predictions':

- Algumas predições não eram de raças caninas;
- Alguns nomes de raças estavam separados por underline;

Quanto a arrumação:

- Este dataframe continha 3 predições da raça do cachorro em colunas diferentes;
- Seria interessante apenas uma coluna com a predição mais confiável.

3) Limpando

Para iniciar o processo de limpeza dos dados, fiz uma cópia de cada dataframe.

Dataframe tweets

- Para solucionar o problema dos numeradores e denominadores, foi extraído do texto a nota do autor. Para isso foi utilizado o padrão RegEx já que a nota seguia o padrão 'numerador / denominador'. Duas funções atribuíram o valor nulo para numeradores e denominadores que não faziam sentido;
- Também com o padrão RegEx, o link no texto dos tweets (que sempre iniciava com ''https://') foi substituída por uma string vazia;
- Com o mesmo padrão, a origem dos tweets foi extraída da coluna 'source';
- Todos os nomes de cachorros que iniciavam com uma letra minúscula foram substituídos pelo texto 'None';
- Ainda na coluna dos nomes, a função 'clean_name' busca as células com o texto 'None' e atribui a ela, de fato, o valor nulo;
- O tipo de dado da coluna 'timestamp' foi mudado para DATETIME;
- Nas colunas do estágio, foi salvo em uma lista os índices das linhas com mais de uma classificação. Em seguida, cada célula dessas colunas recebeu o texto 'None';
- Foram removidas as colunas que não seriam usadas;
- Para o problema da variável 'Estágio' em 4 colunas: foi usado um dataframe temporário com as colunas 'tweet_id', 'doggo', 'floofer', 'pupper' e 'puppo'. Em seguida, com a função 'melt', as colunas 'doggo', 'floofer', 'pupper' e 'puppo' foram dunfidas deixando 'tweet_id' como variável identificadora. Ainda no dataframe temporário, todas as linhas que não tinham uma classificação foram excluídas. Por fim, os dataframes original e temporário foram unidos, excluindo as colunas 'doggo', 'floofer', 'pupper' e 'puppo'.

Dataframe image_predictions

- Foi feita uma consulta para identificar todas as predições que não eram de raças caninas (p1_dog = False, p2_dog=False e p3_dog = False) para serem excluídas;
- O underline no nome de algumas raças foi substituído por um espaço;
- Para selecionar a predição para uma raça canina com maior confiabilidade, uma lista foi gerada com todos os índices das linhas que receberam pelo menos um valor 'False' dentre as colunas p1_dog, p2_dog e p3_dog. Assim, a função 'verifica_raca' troca o valor de uma coluna com o valor True para uma raça canina caso a anterior seja 'False'. Como por exemplo, p3_dog = True e p2_dog = False. Esse processo foi feito com p3 e p2, em seguida com p2 e p1.

Por fim, todos os dataframes limpos foram unidos em apenas um e em seguida guardado em um arquivo CSV chamado 'twitter_archive_master.csv'.