Nanodegree Udacity - Wrangling e Análise de Dados

Relatório Projeto 2 - #WeRateDogs

Por Cleber Cândido

Introdução

Como parte integrante do Nanodegree de Analista de Dados da Udacity, esse projeto de Data Wrangling teve como objetivo reunir dados de uma variedade de fontes e formatos, a saber csv, json, tsv para citar alguns. A fim de avaliar a sua qualidade e organização, e limpá-los. Dessa forma, foi possível obter uma amostra dos dados tratados, através de análises e visualizações.

Os dados foram fornecidos a partir do usuário **@dog_rates**, também conhecido como WeRateDogs, que é uma conta do Twitter que classifica os cachorros a partir de comentários um tanto humorados.

Reunindo os dados de três fontes:

Twitter_archieve_enhanced, o qual contém dados básicos de twitter com mais de 5000 tweets, já disponibilizado no projeto.

Image_predictions.tsv, alguns id´s não receberam uma raça válida da rede neural. Para esse arquivo foi necessário a programação em Python através da URL fornecida.

Tweet_json.txt, Arquivo complementar ao primeiro que precisava ser gerado com base nos id´s do primeiro arquivo utilizando a API do Twitter.

Coletando os dados

Nos dois primeiros arquivos, acima mencionados, foi utilizado o Python para o download através da biblioteca "**request**".

Para o terceiro arquivo, utilizou-se a biblioteca "tweepy" para a API do Twitter, utilizando um arquivo externo para a autenticação na API.

Para tal, se iniciou percorrendo cada id no arquivo "**twitter-archive_enhanced.csv"**, e assim, após importar os dados em DataFrame, obter o arquivo json retornado para cada requisição de id, gravando um arquivo por linha num arquivo nomeado "**tweet_json.txt**".

Análise

1 - visualmente no arquivo **twitter_archive_enhanced**:

- A coluna source possui URLs com tags HTML de link <A>;

- A coluna nome possui parte dos valores com None, e letras minúsculas fora da conformidade,

exemplo: a, actually, all, na, by, etc.

- A coluna rating denominator, possivelmente, com valores fora conformidade, variando entre valores muito baixo e muito altos;

- Nas colunas doggo, floofer etc, os id´s possuíam mais de um valor.

2 - no arquivo **image_predictions.tsv**:

- Alguns id´s não obtiveram uma raça de cachorro da rede neural (p1-dog, p2-dog e p3_dog), estas com valor FALSE ou apenas algumas eram válidas;

- Algumas raças apresentou underscore no nome;

- Um número menor de registros, quando comparado ao arquivo twitter-archive_enhanced.csv.

3 - No arquivo **tweet_json.txt**:

Não foram encontrados problemas no dados.

Limpeza

Foi gerado um único DataFrame para efeito de análise. Como motivação, segundo o projeto, ficou indicado que a análise deveria se dar nos tweets com imagens, assim, se procedeu com o emprego de inner join junto ao processamento das imagens, dessa forma os id´s sem relação estabelecida à image-prediction.tsv foram excluídos. Alguns problemas de qualidade foram resolvidos com expressões regulares.

De uma forma geral, criar algumas colunas como breed e breed_conf, junto à raça, seguida da confiança, foram bastante desafiadoras, assim como pivotear as colunas doggo, floofer, pupper e puppo numa coluna única. Sobre o processo de data wrangling, pareceu muito conveniente seguir o ensinado no curso, Coletar -> Avaliar -> Limpar, dessa forma o entendimento e o resultado se tornou mais claro e objetivo.

Fontes de Estudo

https://stackoverflow.com/questions/28384588/twitter-api-get-tweets-with-specific-id
http://docs.tweepy.org/en/v3.2.0/api.html#API
https://stackoverflow.com/questions/21308762/avoid-twitter-api-limitation-with-tweepy
https://wiki.python.org/moin/ForLoop
https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object.html
https://stackoverflow.com/questions/28056171/how-to-build-and-fill-pandas-dataframe-fr
om-for-loop/28058264
http://stackabuse.com/reading-and-writing-json-to-a-file-in-python/
https://stackoverflow.com/questions/7370801/measure-time-elapsed-in-python
http://docs.tweepy.org/en/v3.2.0/api.html#API
https://stackoverflow.com/questions/28384588/twitter-api-get-tweets-with-specific-id
https://stackoverflow.com/questions/8784396/python-delete-the-words-between-two-deli
miters
https://docs.python.org/3.4/howto/regex.html
http://www.pythonlearn.com/html-007/cfbook012.html