Megadados

Projeto Spark

Edgard Ortiz Neto

Cluster utilizado: *SparkProject*

Tarefa 1)

• Quantos reviews existem?

Número de reviews = 150962278

• Quantos clientes existem?

Número de clientes = 33497620

• Quantos produtos existem?

Número de produtos = 21390118

• Quantos reviews existem para cada “star_rating” (de 1 a 5 estrelas)?

Número de reviews com 1 estrela = 12099424

Número de reviews com 2 estrelas = 7304329

Número de reviews com 3 estrelas = 12133772

Número de reviews com 4 estrelas = 26223155

Número de reviews com 5 estrelas = 93199322

• Item desafio: Quais os 10 produtos que receberam maior rating médio dentre os produtos com mais de 10 ratings, e considerando apenas os ratings de clientes que só deram um review ao todo?

product_id	product_title	count
B009UX2YAC	Subway Surfers	10311
B00E8KLWB4	The Secret Societ…	10023
B0054JZC6E	101-in-1 Games	9370
B00FAPF5U0	Candy Crush Saga	8948
B00G5LQ5MU	Escape The Titanic	8104
B00AREIAI8	My Horse	7949
B004LLIKVU	Amazon.com eGift …	7660
B00992CF6W	Minecraft	7448
B005ZOBNOI	The Fault in Our …	7187
B00QW8TYWO	Crossy Road	6216

Tarefa 2)

Foi observado que há 33497620 de usuários com uma média de aproximadamente 5 reviews para cada usuário, além de um desvio padrão de mais ou menos 20 reviews, portanto, o critério estabelecido, para detectar os bots, foi somar a média com desvio, obtendo-se um número 25 reviews como separador entre humano e bot.

A justificativa para a escolha desse critério se dá pela quantidade estimada de compras e análises que um usuário geralmente faz, pois, uma vez que um “user” ultrapassa a soma da média com desvio, ele pode ser considerado um usuário acima da média da plataforma, que se dá, muito provavelmente, por ou um usuário “hardcore” da Amazon, ou de longa data, ou um bot, já que nesse critério não é esperado que o review seja algo em que um usuário “real” comum faça com frequência.

Por exemplo, esse “user” comum, de todas as compras realizadas na Amazon, ele só vai fazer a avaliação caso ele se sinta muito decepcionado ou muito animado com o produto, caso contrário é pouco provável que o mesmo gaste seu tempo avaliando um produto “normal”.

Logo, a partir de todas essas suposições, mesmo que o mecanismo seja extremamente superficial e ausenta pesquisas para corroborá-lo, foram separados os usuários reais e bots, chegando ao número de 866670 bots (acima de 25 reviews) e 32630950 humanos (25 ou menos reviews), com um máximo de 59623 reviews para um único usuário, no nosso caso, um bot e 49868975 reviews, feitas pelos mesmos, totais.

Além disso, foi observado que os temas onde há mais intervenções de bots são os livros e livros eletrônicos (E-book), com 7000207 e 6783748 avaliações, respectivamente. E eles são bem “bondosos”, com 31266950 avaliações positivas (5 estrelas).

Analisando esses dados, é possível chegar em uma hipótese onde escritores de livros, ou editoras, acabam contratando esse serviço de bot para elevar a avaliação de seus produtos e atraírem mais consumidores.

Tarefa 3)

Para a implementação do modelo *Naive-Bayes*, é melhor observar o arquivo com os outputs do Zeppelin, no caso, SparkProject.html.

Acompanhando a referência (ai.plainenglish.io/build-naive-bayes-spam-classifier-on-pyspark-58aa3352e244), foi necessário remover as linhas preenchidas com “Nulls/Non Available” e criar uma coluna “label_rating” com os significados de cada avaliação, no caso 5 para positivo, 4 para neutro e 3 ou menos como negativo, além de especificar as colunas com o conteúdo (“review_body”) e o label (“label_rating”) das avaliações.
Após o treino do modelo, foi feita a implementação que obteve acurácia de aproximadamente 74% (0.7417851034134614).