

Universidade Católica Dom Bosco
Curso de Engenharia da Computação
Disciplina de Visão computacional

Título do Trabalho: Análise de Machine Learning em Pokémons

Nome do Aluno: Raphael Victor

Professor: Hemerson Pistori

Data: 28 de maio de 2024

Primeiramente, gostaria de contextualizar o meu experimento e o que eu utilizei para o machine learning. Para o projeto, eu usei pokémons. Mas por que pokémons? Porque existem ao todo 18 tipos. Então, eu escolhi 6 deles para o experimento, sendo eles: Água, Grama, Fogo, Dragão, Fada e Sombrio. Além disso, há outro motivo pelo qual escolhi os pokémons para utilizar no experimento: cada um deles tem sua própria característica principal, que é ditada pelo seu tipo e pelo ambiente em que vive.

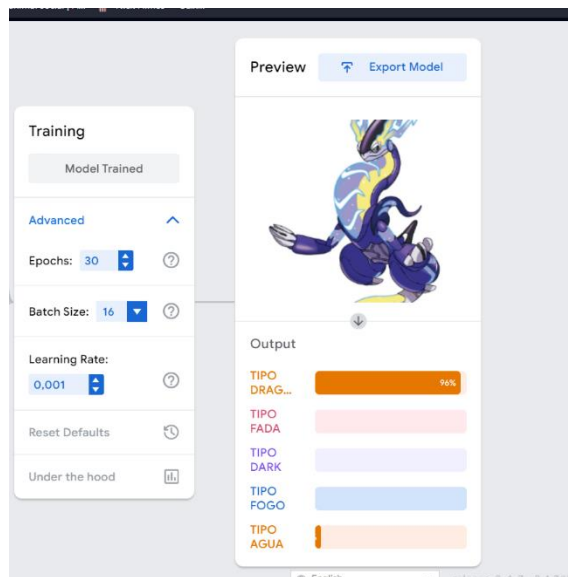


Em Pokémon, existem ao todo 9 gerações. Então, eu fiz da seguinte forma: selecionei pokémons da 1ª à 7ª geração para ensinar à máquina o que era e o que caracterizava tal tipo. Para os experimentos, eu usei pokémons da 8ª e 9ª geração.

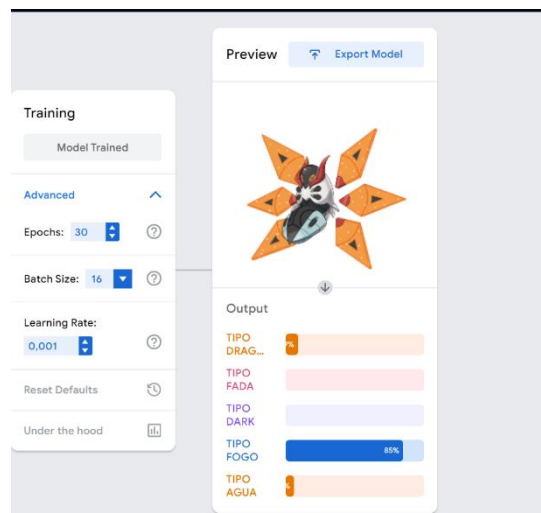
Na passo a seguir, eu comecei a experimentar com alguns pokémons mais óbvios, que certamente seriam daqueles tipos.



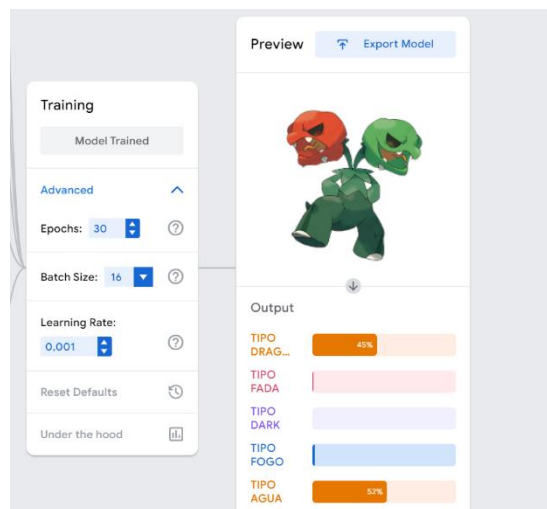
Nome: Miraidon
Nº: 1008
Tipo: Elétrico/Dragão
Evolui: Lendário



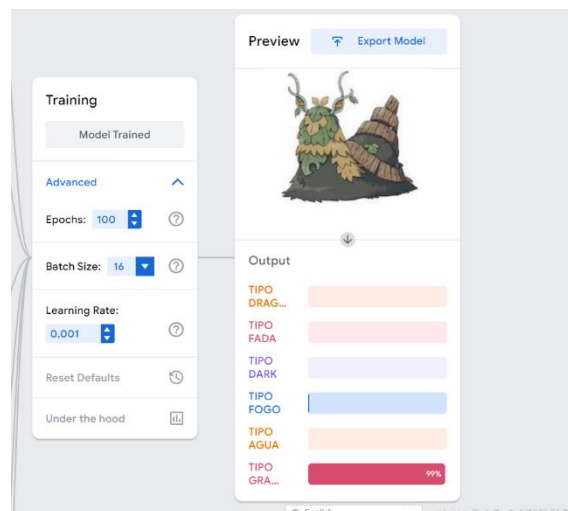
Nome: Iron Moth
Nº: 994
Tipo: Fogo/Veneno
Evolui: Paradox



Ambos acima foram feitos utilizando 30 de epochs até que com essa mesma quantia começou a dar errado como no exemplo a seguir;



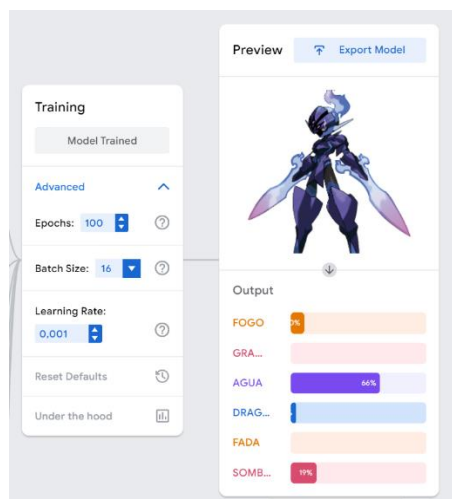
E foi nesse momento em que eu decidi começar a aumentar a epochs para aumentar o número de vezes que todo o conjunto de dados de treinamento é mostrado à rede durante o treinamento;



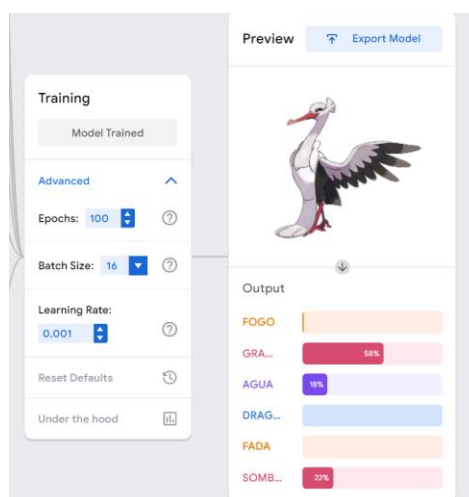
Mesmo aumentando as epochs nos dois experimentos seguintes, nossa máquina falhou em tentar adivinhar os tipos dos pokémons.



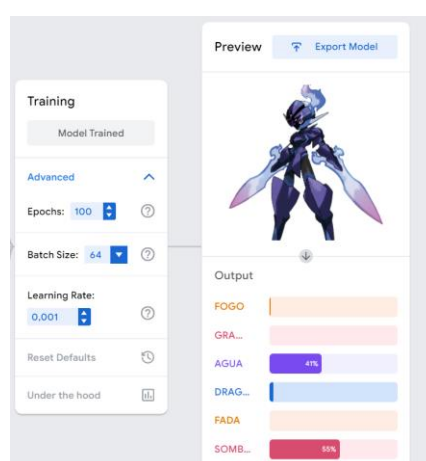
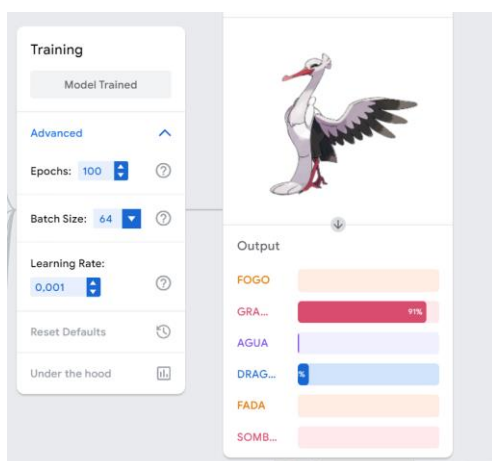
Nome: Ceruleedge
Nº: 937
Tipo: Fogo/Fantasma
Evolui: Não



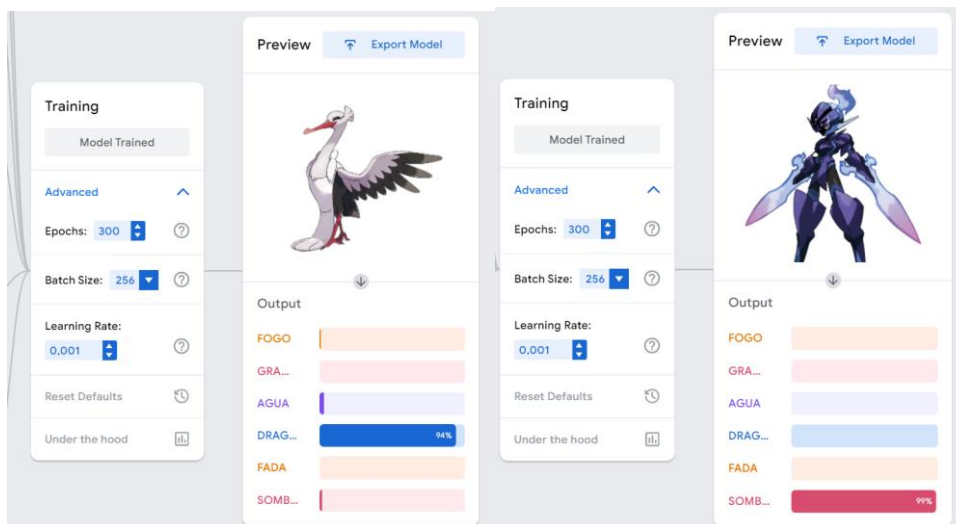
Nome: Bombirdier
Nº: 962
Tipo: Voador/Noturno
Evolui: Não



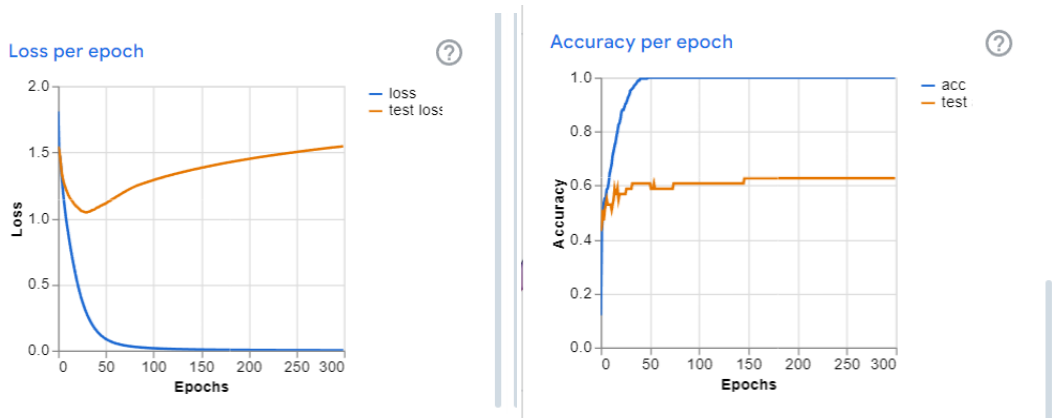
Agora, para o próximo passo, eu tentei realizar os mesmos dois experimentos anteriores, mas aumentando o batch size para que o lote de pesquisa dentro de nossa máquina fosse maior.



Nos dois experimentos acima, ainda tivemos resultados consideravelmente ruins. No entanto, um deles apresentou uma melhora significativa: o nosso Cerulede, que é dos tipos Fogo e Sombrio. Antes, nenhum dos atributos foi identificado corretamente, já agora o mesmo já mostrou indícios de possuir suas verdadeiras tipagens. Então, para tentar melhorar os resultados, aumentaremos ainda mais o batch size e as epochs e aplicaremos nos mesmos dois pokémons.



No experimento acima, não aumentamos o learning rate. Portanto, apenas pelo aumento dos hiperparâmetros houve um aumento significativo de resultados, mas ainda assim, não é conclusivo.



Loss:

No início, a perda de treinamento (curva azul) diminui rapidamente, indicando que o modelo está melhorando seu ajuste aos dados de treinamento. A perda de validação (curva laranja) inicialmente também diminui, mas após um certo ponto, ela começa a aumentar novamente, enquanto a perda de treinamento continua a diminuir. Esse comportamento é outro indicativo de overfitting, pois o modelo está se ajustando tão bem aos dados de treinamento que perde a capacidade de prever corretamente os dados de validação.

Accuracy:

No início, a acurácia de treinamento (curva azul) aumenta rapidamente e atinge valores próximos de 1 (ou 100%), indicando que o modelo está aprendendo bem no conjunto de treinamento. A acurácia de validação (curva laranja) também aumenta inicialmente, mas, após um certo ponto, ela se estabiliza e não melhora mais, permanecendo significativamente mais baixa que a acurácia de treinamento. Isso sugere que o modelo pode estar sofrendo de overfitting, ajustando-se muito bem aos dados de treinamento, mas não generalizando bem para novos dados (dados de teste).

Accuracy per class



CLASS	ACCURACY	# SAMPLES
FOGO	1.00	8
GRAMA	0.83	12
AGUA	0.50	18
DRAGÃO	0.00	5
FADA	0.67	3
SOMBRIO	0.60	5

O modelo tem um desempenho muito bom para a classe "FOGO", com 100% de acurácia. O desempenho é razoável para as classes "GRAMA", "FADA" e "SOMBRIO", mas ainda há espaço para melhorias. No entanto, o modelo enfrenta dificuldades significativas com as classes "ÁGUA" e "DRAGÃO", especialmente com "DRAGÃO", onde não conseguiu classificar nenhuma amostra corretamente. A quantidade de amostras varia bastante entre as classes, o que pode influenciar na acurácia reportada. Classes com poucas amostras, como "FADA" e "DRAGÃO", podem ter uma avaliação de desempenho menos confiável.

Agora, para tentar melhorar os resultados com base nas análises acima, aumentaremos o batch size, diminuiremos as epochs e aumentaremos o learning rate. Veremos como nossa máquina vai se comportar com os mesmos dois pokémons dos exemplos acima.

Nosso resultado mais interessante ocorreu quando aumentamos as epochs e o learning rate. Porque, pelo menos, nosso Ceruledge agora apresentou indícios de ser do tipo Fogo e foi classificado como tipo Sombrio, sem nenhuma interferência de outras tipagens.

Concluimos que nosso modelo tem muita dificuldade em capturar subtipagens que não estão explícitas em nosso personagem. Tentei melhorar os resultados de todas as maneiras possíveis, e esse foi o mais próximo que consegui chegar do ideal.