

# DOG-BREED-RECOGNITION-V3



## A Solução

O sistema de reconhecimento de raças de cachorro foi desenvolvido e testado em:

- Python 3.7, Pytorch, Torch 1.7.1, Torchvision 0.8.2
- Jupyter Notebook e GPUs do Kaggle
- Para informações sobre ambiente completo, por favor consulte:  
<https://github.com/Kaggle/docker-python/blob/b58e00f4ef903abafb4b04b957744c4d07c9c551/gpu.Dockerfile>
- Utility script: [https://github.com/amandaleonel/dog-breed-recognition-v3/blob/main/recgn\\_utils.py](https://github.com/amandaleonel/dog-breed-recognition-v3/blob/main/recgn_utils.py)

## Primeira parte

### Dogs: Sistema de detecção de raças de cachorros

- Total de classes: 100
- Total de imagens: 17.345



- Transfer learning: Fine tuning
- Modelo pré-treinado: Resnet152
- Classificador contendo camada LogSoftmax
- Otimizador SGD (stochastic gradient descent)
- Loss function: NLLLoss (negative log likelihood loss)
- Modelo obtido: Epoch = 15; Validation loss = 0.52; Acurácia = 83%

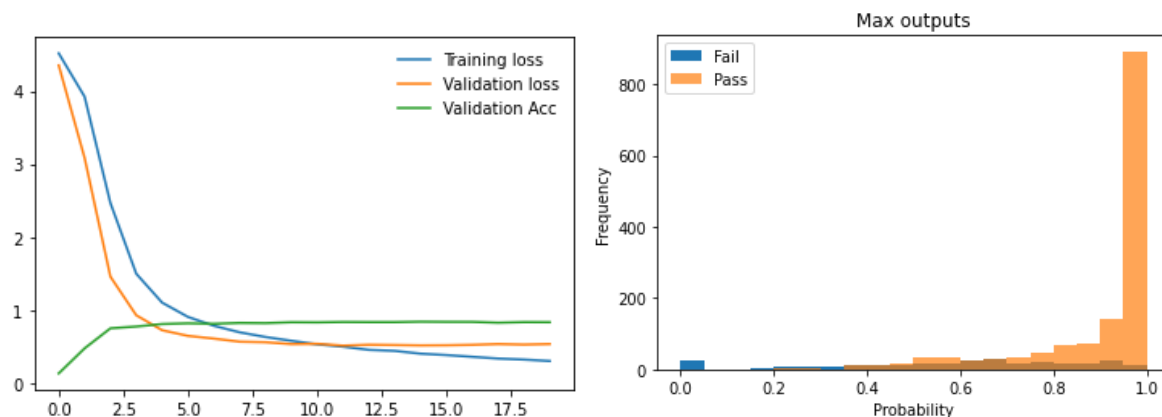


Figura 1a. Evolução das perdas e acurácia, ao longo das épocas de treinamento e validação.

Figura 1b. Medida de probabilidade na camada de saída da rede durante testes.

Logs e notebooks: [https://github.com/amandaleonel/dog-breed-recognition-v3/blob/main/part-1/model\\_epoch\\_15\\_acc\\_84.4318\\_loss\\_0.5199.ipynb](https://github.com/amandaleonel/dog-breed-recognition-v3/blob/main/part-1/model_epoch_15_acc_84.4318_loss_0.5199.ipynb)

## Segunda parte

**Enroll:** Sistema de detecção de novas raças de cachorros

- Total de classes: 20
- Total de imagens: 2.240

n02107683-Bernese\_mountain\_dog



n02087394-Rhodesian\_ridgeback



n02109961-Eskimo\_dog



n02105056-groenendael



- Transfer learning: Feature extraction
- Modelo pré-treinado: Sistema da Primeira Parte (Dogs)
- Classificador contendo camada LogSoftmax
- Otimizador SGD (stochastic gradient descent)
- Loss function: NLLLoss (negative log likelihood loss)
- Modelo obtido: Epoch = 25; Validation loss = 0.13; Acurácia = 95%

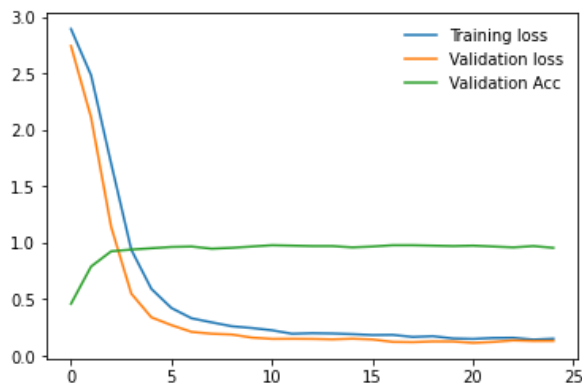


Figura 2a. Evolução das perdas e acurácia, ao longo das épocas de treinamento e validação.

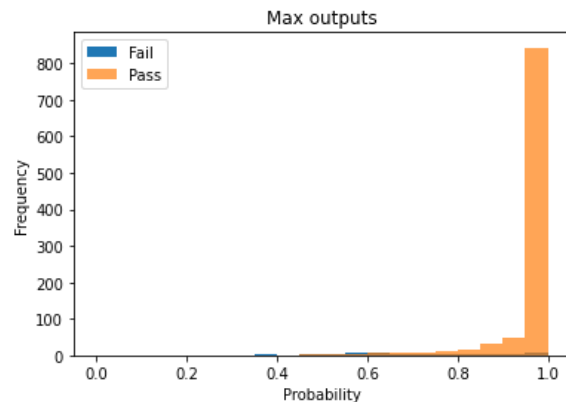
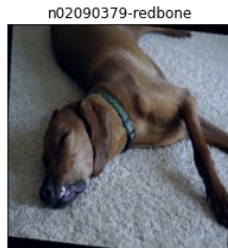


Figura 2b. Medida de probabilidade na camada de saída da rede durante testes.

Logs e notebooks: [https://github.com/amandaleonel/dog-breed-recognition-v3/blob/main/part-2/model\\_20cl\\_epoch\\_25\\_acc\\_95.3846\\_loss\\_0.1295.ipynb](https://github.com/amandaleonel/dog-breed-recognition-v3/blob/main/part-2/model_20cl_epoch_25_acc_95.3846_loss_0.1295.ipynb)

### Enroll: Sistema de detecção de novas raças de cachorros

- Total de classes: 5
- Total de imagens: 55



- Transfer learning: Feature extraction
- Modelo pré-treinado: Sistema da Primeira Parte (Dogs)
- Classificador contendo camada LogSoftmax
- Otimizador SGD (stochastic gradient descent)
- Loss function: NLLLoss (negative log likelihood loss)
- Modelo obtido: Epoch = 24; Validation loss = 0.78; Acurácia = 80%

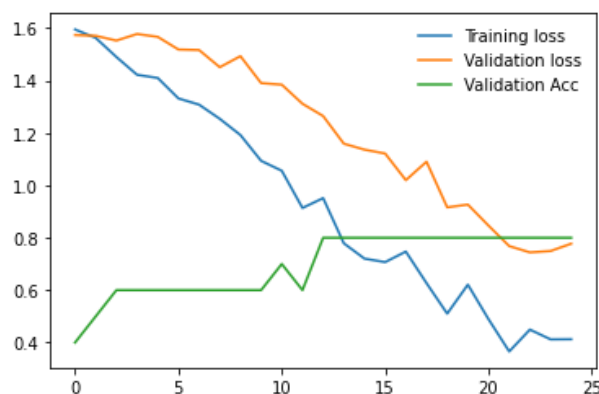


Figura 3a. Evolução das perdas e acurácia, ao longo das épocas de treinamento e validação.

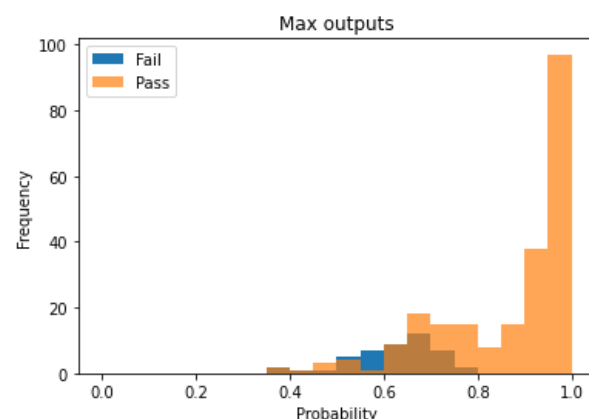


Figura 3b. Medida de probabilidade na camada de saída da rede durante testes.

Logs e notebooks: [https://github.com/amandaleonel/dog-breed-recognition-v3/blob/main/part-2/model\\_5cl\\_epoch\\_25\\_acc\\_80.0000\\_loss\\_0.7775.ipynb](https://github.com/amandaleonel/dog-breed-recognition-v3/blob/main/part-2/model_5cl_epoch_25_acc_80.0000_loss_0.7775.ipynb)

## Terceira parte

### Unknowns: Análise de detecção de raças desconhecidas

- Total de imagens Unknowns apresentadas: 17.345
- Modelos utilizados: Sistemas da Segunda Parte (Enroll) com 5 e 20 classes treinadas.

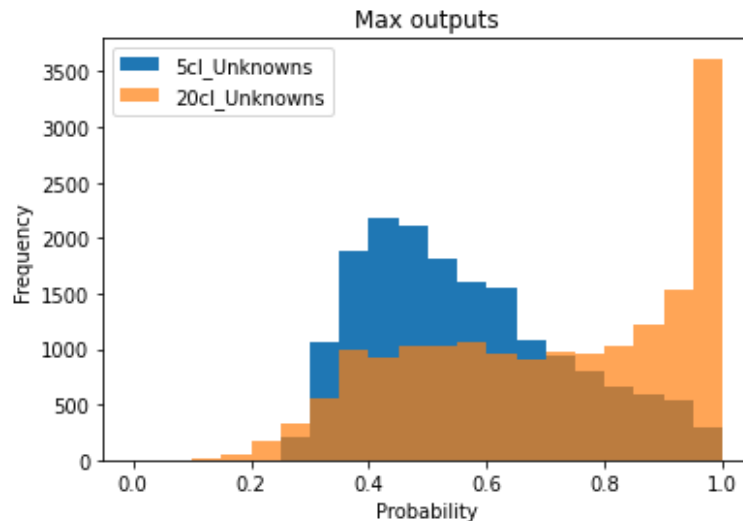


Figura 4. Medida de probabilidade na camada de saída das redes durante apresentação dos **unknowns**.

Logs e notebooks: <https://github.com/amandaleonel/dog-breed-recognition-v3/blob/main/part-3/dog-breed-recognition-part3.ipynb>

O problema de identificação de unknowns por vezes é comparado na literatura aos problemas de *open set recognition* e *one-class classification* dada a dificuldade em não apenas classificar corretamente em uma ou mais classes conhecidas, mas também de rejeitar corretamente classes desconhecidas.

Analisando algumas medidas de probabilidade na camada de saída das redes durante apresentação de knowns e unknowns, é possível observar que modelos de alta acurácia tendem a responder menos ativamente a imagens unknowns, isto é, com menores probabilidades na saída predita. Se a probabilidade de classificação não é boa suficiente, o modelo pode estar lidando com uma classe desconhecida.

A aplicação de um limiar nas probabilidades de saída, diretamente proporcional à acurácia do modelo testado – para descartar classificações com saídas abaixo desse limiar – pode trazer benefícios quanto à rejeição de unknowns, mas levando também ao descarte de falsos e verdadeiros positivos. Idealmente mais experimentos poderiam ser realizados com ajuda de outros classificadores aplicados à camada de saída.