

Equipe - Classificação de Animais

Alunos:

- **Julio Leocadio de Oliveira Falcão**
- **Gisele Melio de Oliveira**
- **Kesava Yasmim Silva de Menezes**
- **Luan Fernandes Alves**
- **João Marcos Rodrigues da Silva**

Sumário

- Introdução
- Metodologia
- Resultados
- Conclusões

Introdução

Resumo:

Classificação de Animais usando Redes Neurais Convolucionais (CNNs).

Objetivo Geral:

Apresentar métodos da literatura e métodos próprios.

Introdução ao Dataset

Metadata:

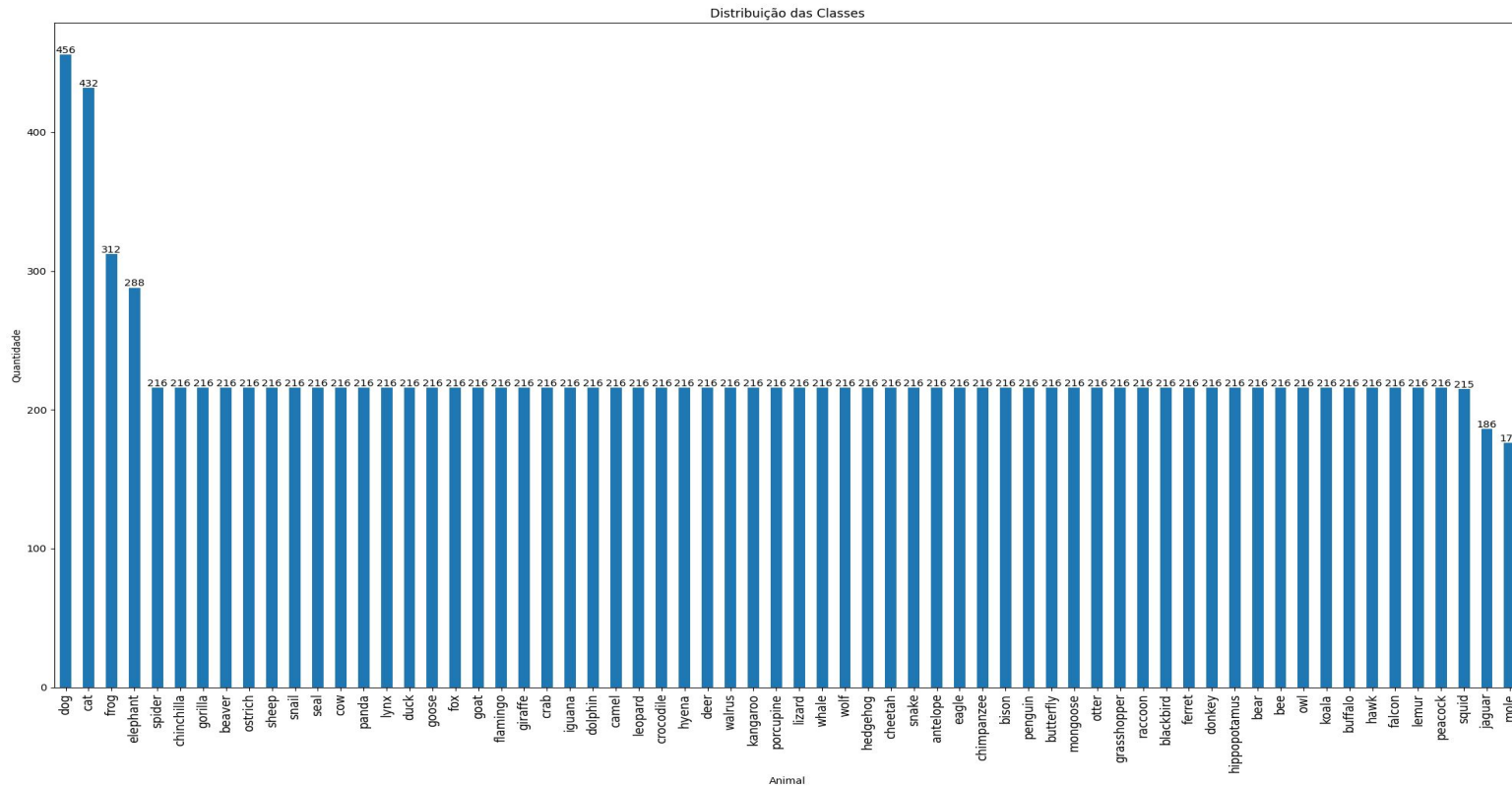
Classes: 64 classes de animais;

Número de Imagens: 14377;

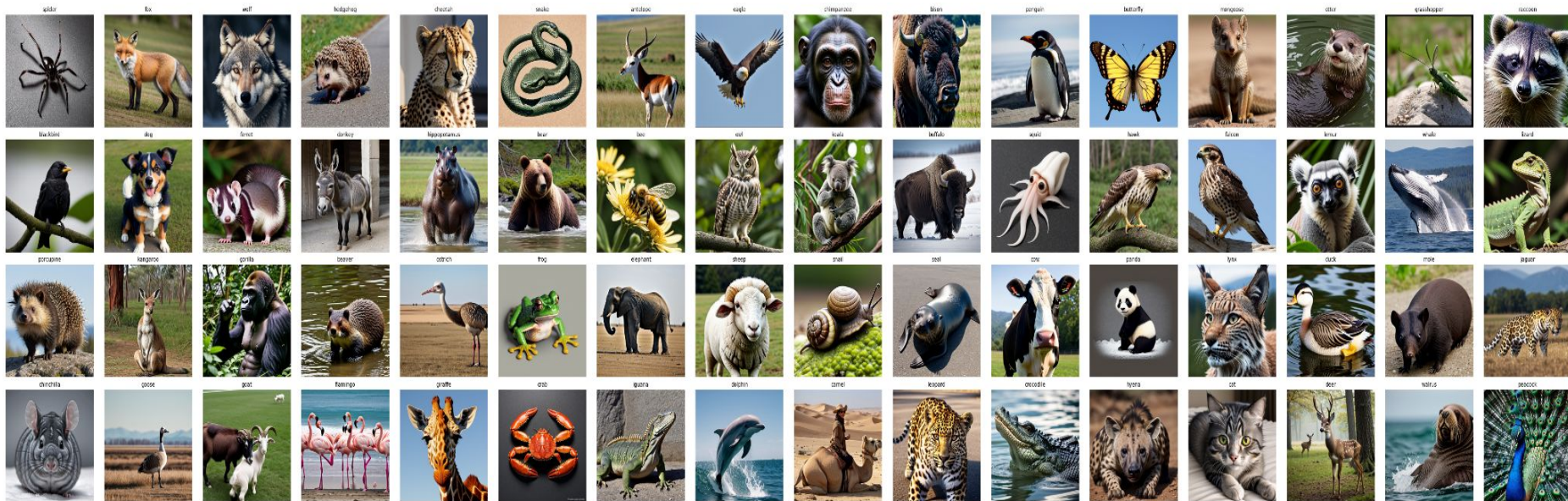
Tamanho: 512 x 512 pixels;

Imagens Corrompidas: 0;

Introdução ao Dataset



Introdução ao Dataset



Introdução ao Notebook (Literatura)

Transformações:

- **Redimensionamento:** (224, 224) pixels;
- Inversão horizontal aleatória;
- Normalização dos valores dos pixels para o padrão do ImageNet:
 - **Média:** [0.485, 0.456, 0.406]
 - **Desvio padrão:** [0.229, 0.224, 0.225]

Split:

- 70% Treinamento;
- 15% Validação;
- 15% Teste;

Introdução ao Notebook (Literatura)

Tamanho do Lote: 64

Modelo: ResNet152 pré-treinado

Função de perda: Cross Entropy Loss

Otimizador: Adam

Tx. de Aprendizado: 0.0001

Termo de Regularização L2: 0.001

Épocas: 20

Resultados:

- **Train Loss:** 0.0199
- **Train Accuracy:** 99.66%
- **Validation Loss:** 0.0269
- **Validation Accuracy:** 99.40%
- **Test Accuracy:** 99.12%
- **Precision:** 99.22%
- **Recall:** 99.12%
- **F1-Score:** 99.10%

Metodologia

Transformações:

- **Redimensionamento:** **(224, 224)** pixels;
- **Brilho:** **0.3** – A imagem pode ter um brilho reduzido até 70% ($1 - 0.3$) ou aumentado até 130% ($1 + 0.3$) do valor original;
- **Contraste:** **0.2** – O contraste será ajustado dentro do intervalo de 80% ($1 - 0.2$) a 120% ($1 + 0.2$) do valor original;
- **Saturação:** **0.2** – A saturação será ajustada dentro do intervalo de 80% ($1 - 0.2$) a 120% ($1 + 0.2$) do valor original;
- **Rotação:**
 - **Vertical e Horizontal:** **(p = 0.5)** – para cada imagem, há 50% de chance de que ela seja virada verticalmente ou horizontalmente. Caso contrário, a imagem permanece como está.

Metodologia

Transformações:

- **Corte Aleatório:**
 - **Escala: (0.8, 1.0)** – Define que a área da imagem a ser cortada será entre 80% a 100% do tamanho original;
 - **Razão: (0.9, 1.1)** – O aspecto da imagem cortada será ajustado aleatoriamente, com a relação entre largura e altura da região cortada variando entre 0.9 e 1.1
- Normalização dos valores dos pixels para o padrão do ImageNet:
 - **Média: [0.485, 0.456, 0.406]**
 - **Desvio padrão: [0.229, 0.224, 0.225]**

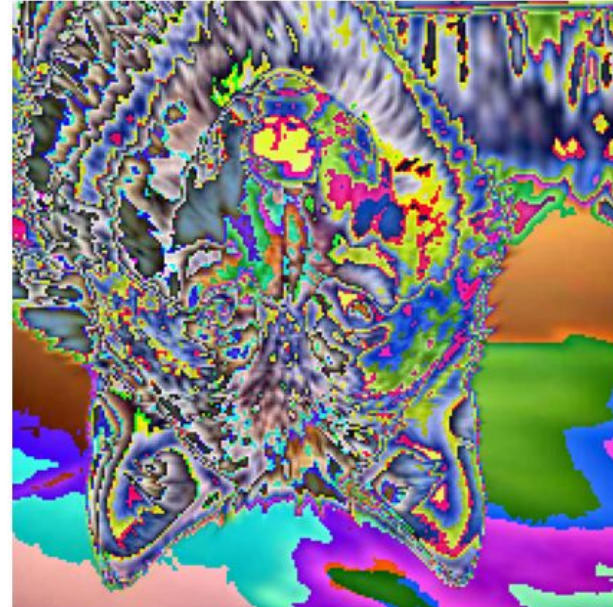
Metodologia

Exemplo:

Imagem Original



Imagem Transformada



Metodologia

Exemplo:

Imagem Original



Imagem Transformada



Metodologia

Split:

- 70% Treinamento;
- 15% Validação;
- 15% Teste.

Modelo: **EfficientNet_B2** pré-treinado

Total params: 7,791,170

```
Default: Sequential(  
  (0): Dropout(p=0.3, inplace=True)  
  (1): Linear(in_features=1408, out_features=1000, bias=True)  
)
```

```
CustomModel: Sequential(  
  (0): Dropout(p=0.3, inplace=True)  
  (1): Linear(in_features=1408, out_features=64, bias=True)  
)
```

Metodologia

Tamanho do Lote: 16

Épocas: 50

Paciência: 10

Otimizador: Adam

Tx. de Aprendizado: 0.001

Trm. de Regularização L2: 0.001

Dropout: 0.3

Função de Perda: Cross Entropy Loss

Paciência da Tx. de Aprendizado: 5

Metodologia

- **Acurácia**
- **Perda(Loss)**
- **Recall**
- **F1-Score**
- **Matriz de Confusão**

Resultados

Versão 1.0

Último modelo:

Epoch 26/50

Train Loss: 0.0049 | **Train Accuracy:** 99.91%

Validation Loss: 0.0153 | **Validation Accuracy:** 99.68%

Learning Rate: 1e-05

Validation accuracy improved. Saving model with val_loss

0.015295...

Versão 2.0

Último modelo:

Epoch 47/50

Train Loss: 0.0032 | **Train Accuracy:** 99.94%

Validation Loss: 0.0082 | **Validation Accuracy:** 99.86%

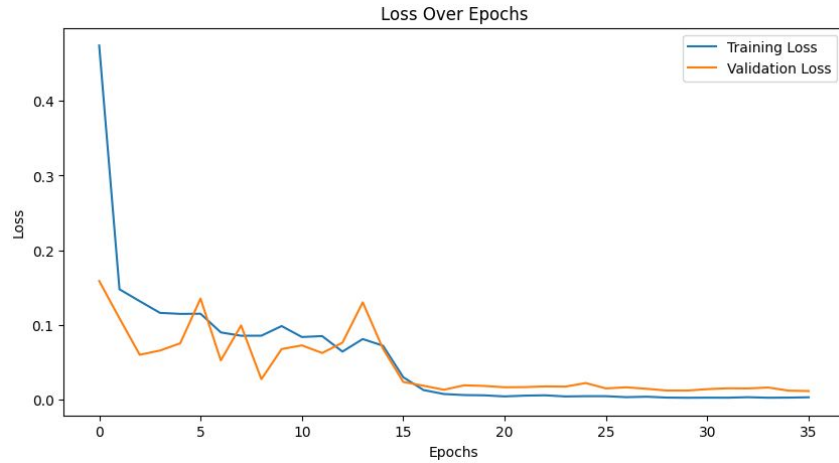
Learning Rate: 1.0000000000000002e-07

Validation accuracy improved. Saving model with val_loss

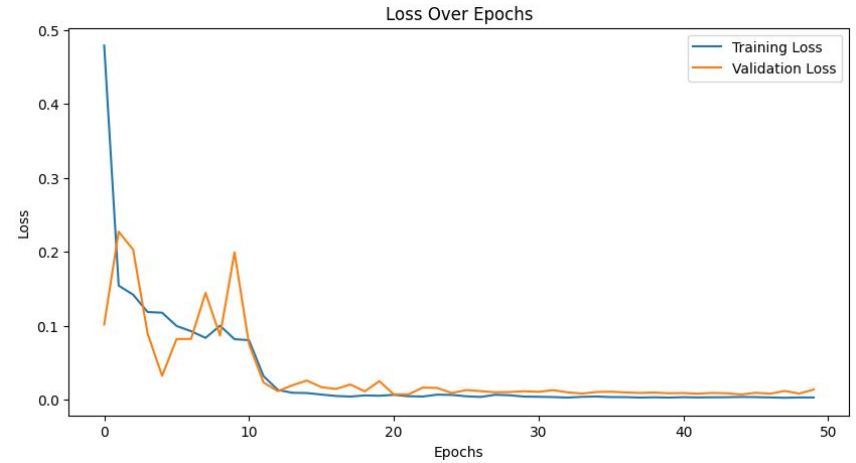
0.008185...

Resultados

Versão 1.0

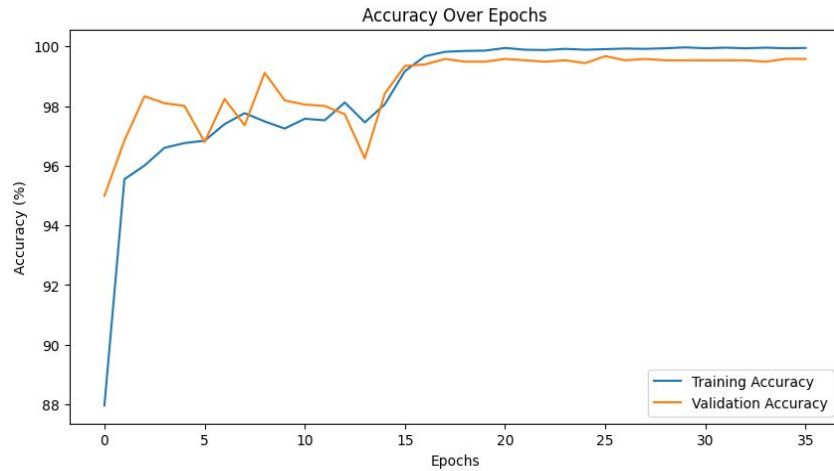


Versão 2.0

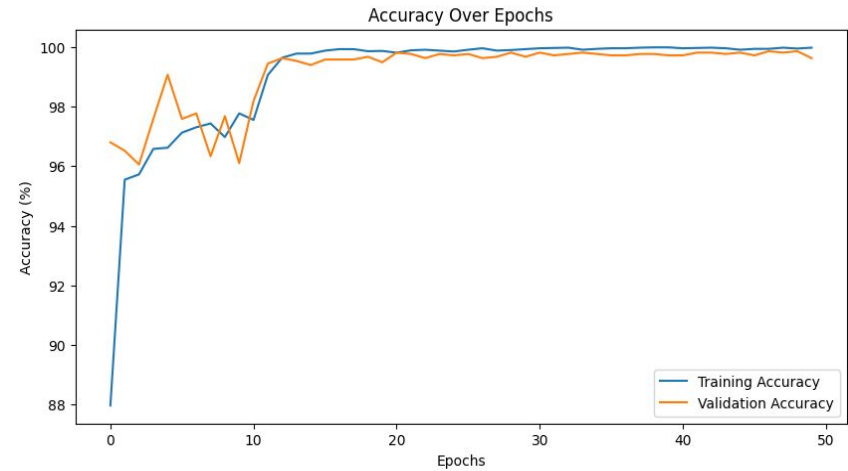


Resultados

Versão 1.0



Versão 2.0



Resultados

Versão 1.0

Total Samples: 2158

Correct Predictions: 2156

Test Loss: 0.003845

Accuracy: 99.91%

Recall: 0.999073

F1 Score: 0.999072

Versão 2.0

Total Samples: 2158

Correct Predictions: 2154

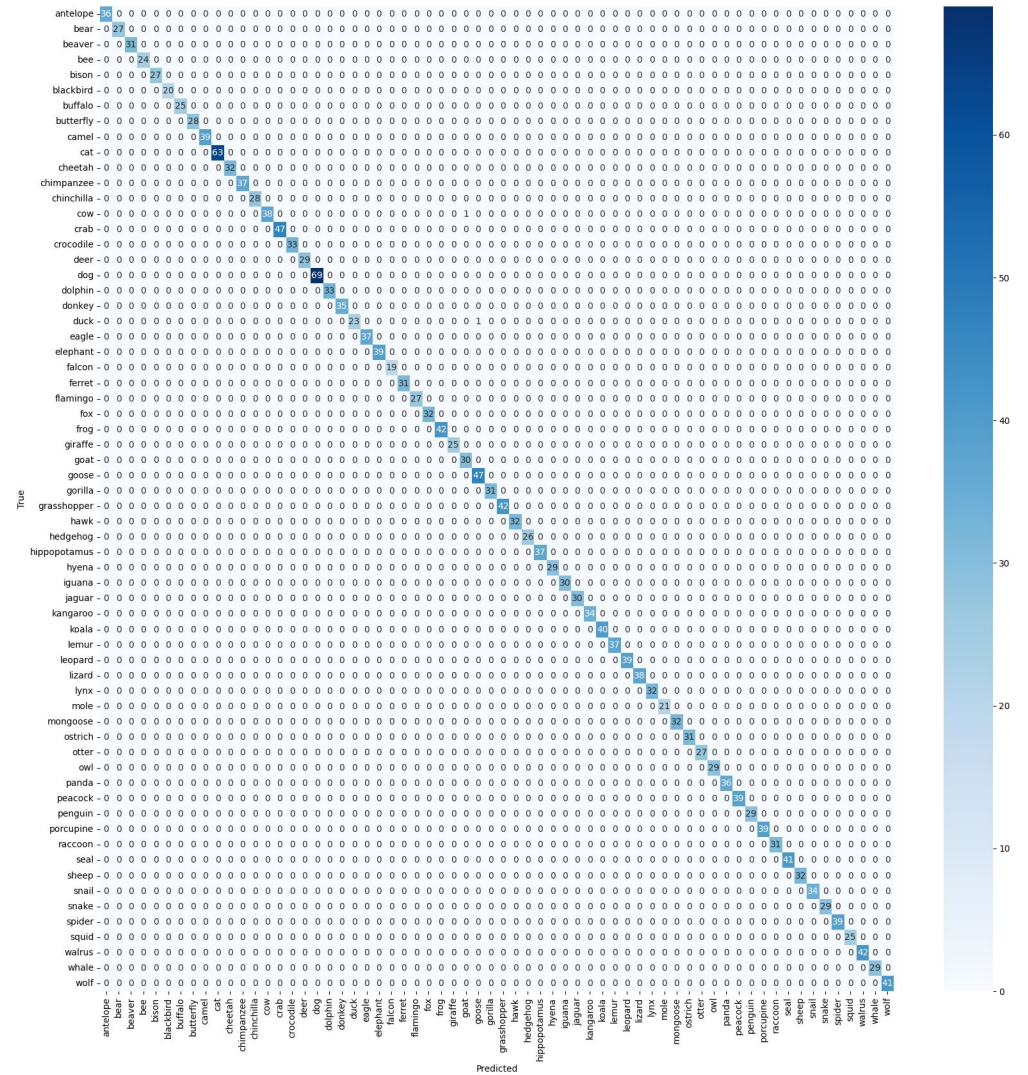
Test Loss: 0.008125

Accuracy: 99.82%

Recall: 0.998146

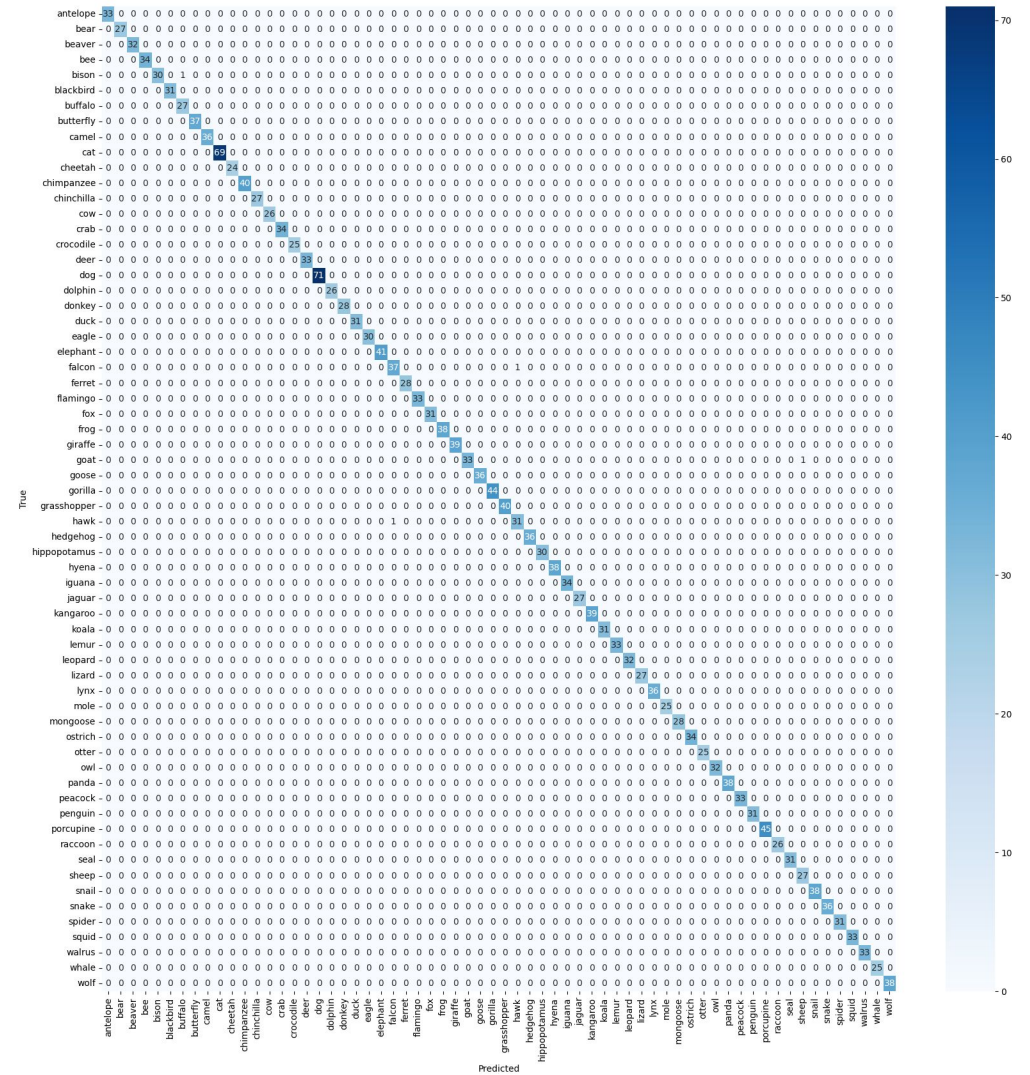
F1 Score: 0.998148

Versão 1.0



Resultados

Versão 2.0



Conclusões

Versão “Literatura”

Train Loss: **0.0199**

Train Accuracy: **99.66%**

Validation Loss: **0.0269**

Validation Accuracy: **99.40%**

Test Accuracy: **99.12%**

Precision: **99.22%**

Recall: **99.12%**

F1-Score: **99.10%**

Versão 2.0

Train Loss: **0.0032**

Train Accuracy: **99.94%**

Validation Loss: **0.0082**

Validation Accuracy: **99.86%**

Test Loss: **0.008125**

Accuracy: **99.82%**

Recall: **0.998146**

F1 Score: **0.998148**

Conclusões

- O modelo **EfficientNet-B2** foi mais eficiente na tarefa, apesar de ter um número significativamente menor de parâmetros e camadas;
- Abre-se caminho para tentar tornar o modelo mais generalista, mantendo a eficiência, através da combinação de com outros modelos menores, como o **ResNet50**, **ResNet34** ou **ConvNeXt_Tiny**;

Conclusões

- Os dados utilizados são **sintéticos**, gerados por IA, o que implica que podem não refletir completamente a complexidade e as variações dos dados do mundo real. Seria sensato expor o modelo a dados reais.

Exemplos:



Referências Bibliográficas

TAN, Mingxing; LE, Quoc V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning (ICML), 2019. p. 6105–6114.

HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. Deep Residual Learning for Image Recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016. p. 770–778.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Yoshua. Deep learning. Cambridge, MA: MIT Press, 2016.

SENGUPTA, A. et al. Synthesizing informative data via generative models: A review. IEEE Access, v. 8, p. 35432–35453, 2020.