Relatório Lista 12: Redes Neurais Convolucionais (CNN)

Douglas Nicolas Silva Gomes

8 de junho de 2025

Conteúdo

1 Introdução **1**

2 Pré-processamento dos Dados **2**
2.1 Organização dos Dados 2
2.2 Técnicas Aplicadas 2

3 Arquitetura da CNN **2**
3.1 Configuração do Modelo 2

4 Compilação e Treinamento **3**
4.1 Configurações 3

5 Melhor Modelo Obtido **3**
5.1 Desempenho 3
5.2 Gráficos de Treinamento 4

6 Avaliação no Conjunto de Teste **4**
6.1 Métricas Quantitativas 4
6.2 Métricas Quantitativas 4
6.3 Predições Externas 5

7 Conclusão **6**

1 Introdução

Este relatório documenta o desenvolvimento de um modelo de classificação binária para distinguir entre imagens de cachorros e gatos utilizando Redes Neurais Convolucionais (CNN). O trabalho foi implementado com TensorFlow/Keras e segue o pipeline completo: pré-processamento, construção do modelo, treinamento e avaliação.

2 Pré-processamento dos Dados

2.1 Organização dos Dados

O dataset foi organizado em três conjuntos:

- Treino (70%)
- Validação (15%)
- Teste (15%)

2.2 Técnicas Aplicadas

- Redimensionamento para 128x128 pixels
- Normalização (valores de pixel entre 0 e 1)
- Data augmentation (apenas para treino):
  - Rotação aleatória (20°)
  - Zoom aleatório (20%)
  - Flip horizontal

```
train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
)
```

3 Arquitetura da CNN

3.1 Configuração do Modelo

- 3 camadas Conv2D + MaxPooling2D
- 1 camada Dense com Dropout (50%)
- Saída binária com ativação sigmoid

```
model = Sequential([
    Conv2D(32, (3,3), activation='relu', input_shape=(128,128,3)),
    MaxPool2D((2,2)),
    Conv2D(64, (3,3), activation='relu'),
    MaxPool2D((2,2)),
    Conv2D(128, (3,3), activation='relu'),
    MaxPool2D((2,2)),
    Flatten(),
```

```
    Dense(256, activation='relu'),
    Dropout(0.5),
    Dense(1, activation='sigmoid')
])
```

4 Compilação e Treinamento

4.1 Configurações

- Otimizador: Adam
- Função de perda: Binary Crossentropy
- Métrica: Acurácia
- Batch size: 32
- Épocas: 50 (com EarlyStopping)

```
early_stop = EarlyStopping(
    monitor='val_loss',
    patience=5,
    restore_best_weights=True
)
```

5 Melhor Modelo Obtido

5.1 Desempenho

- Acurácia de validação: 81.86%
- Loss de validação: 0.4396
- Épocas de treinamento: 23 (com early stopping)

5.2 Gráficos de Treinamento

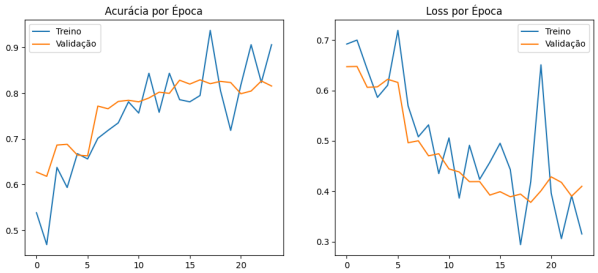

Figura 1: Evolução da acurácia e loss durante o treinamento

6 Avaliação no Conjunto de Teste

6.1 Métricas Quantitativas

6.2 Métricas Quantitativas

Classe	Precisão	Recall	F1-Score	Suporte
Cats	0.82	0.80	0.81	1011
Dogs	0.81	0.83	0.82	1012
Acurácia Total		0.82 (2023 amostras)		
Macro Média	0.82	0.82	0.82	2023
Média Ponderada	0.82	0.82	0.82	2023

Tabela 1: Métricas de desempenho no conjunto de teste

Detalhes adicionais:

- Acurácia final no teste: **0.8151**
- Loss final no teste: **0.4225**
- Tempo por etapa: ˜3s, 49ms/step

6.3 Predições Externas

Imagem	Predição	Confiança
gato1.jpg	Cachorro (Incorreta)	64.58%
gato2.jpg	Gato (Correta)	81.41%
gato3.jpg	Gato (Correta)	98.34%
cachorro1.jpg	Cachorro (Correta)	98.10%
cachorro2.jpg	Cachorro (Correta)	92.68%
cachorro3.jpg	Cachorro (Correta)	98.63%

Tabela 2: Resumo das predições externas com respectivas confianças

Figura 2: Imagem: gato1.jpg - Predição incorreta como Cachorro (Confiança: 64.58%)

Figura 3: Imagem: gato2.jpg - Predição correta como Gato (Confiança: 81.41%)

Figura 4: Imagem: gato3.jpg - Predição correta como Gato (Confiança: 98.34%)

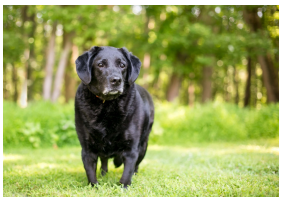

Figura 5: Imagem: cachorro1.jpg - Predição correta como Cachorro (Confiança: 98.10%)

Figura 6: Imagem: cachorro2.jpg - Predição correta como Cachorro (Confiança: 92.68%)

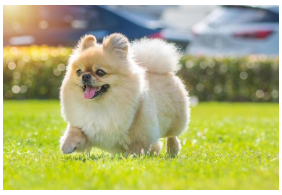

Figura 7: Imagem: cachorro3.jpg - Predição correta como Cachorro (Confiança: 98.63%)

7 Conclusão

O modelo desenvolvido alcançou:

- Boa capacidade de generalização (81.51% de acurácia)
- Balanceamento entre as classes (F1-score de 0.81-0.82)
- Alta confiança na maioria das predições corretas (¿90%)
- O tempo médio de predição foi de 75ms por imagem
- O modelo classificou incorretamente a imagem gato1.jpg como cachorro