



ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO
DEPARTAMENTO DE ENGENHARIA DE SISTEMAS ELETRÔNICOS

RELATÓRIO EXERCÍCIO PROGRAMA PSI5790

NOME

No. USP

JEAN CARLOS MELLO XAVIER FARIA

11259628

São Paulo
2024

SUMÁRIO

1. INTRODUÇÃO.....	2
2. PREPARAÇÃO DO AMBIENTE E DOS DADOS.....	3
3. CLASSIFICAÇÃO - REDE NEURAL TREINADA DO ZERO.....	4
3.1. Treinamento.....	6
3.2 Teste e Curva ROC.....	6
4. CLASSIFICAÇÃO - REDE NEURAL COM TRANSFER LEARNING.....	7
4.1 Treinamento congelando as camadas-base.....	8
4.2 Fine Tuning.....	9
4.3 Teste e curva ROC.....	9
5. CONCLUSÃO.....	10
REFERÊNCIAS.....	11
ANEXOS.....	12

1. INTRODUÇÃO

No final do ano de 2019 um novo vírus conhecido como SARS-Cov-2 surgiu matando milhões de pessoas ao redor do mundo. A doença denominada como COVID-19 afetou pessoas ao redor do mundo. Por se tratar de uma doença pulmonar as pessoas precisavam tirar raio-x para entender o quão grave o pulmão foi afetado pela doença.

Devido a proporção da doença, diversas imagens de raio-x de pessoas com COVID-19 foram tiradas e criado um conjunto de dados com diversos raio-x de pulmão com a doença e sem ela. O conjunto de dados COVID-QU-Ex publicado na plataforma Kaggle organizou todas essas imagens, as rotulando como COVID-19, não COVID-19 (infecção viral ou bacteriana) e normal. As imagens do dataset são imagens de tamanho 256x256 pixels no formato .png.

O objetivo deste relatório é desenvolver um programa para classificar as imagens de raio-x em COVID-19 e não COVID-19 (não COVID-19 e normal). Para resolver essa tarefa, será criada uma rede neural do zero, sendo treinada com um conjunto de dados de treino/validação e testada com um conjunto de imagens de teste. A segunda parte será criada uma rede neural com transfer learning, passando por um treinamento congelando as camadas da rede e depois realizando um treinamento com fine tuning. Por fim, será realizada uma comparação entre ambos os modelos.

2. PREPARAÇÃO DO AMBIENTE E DOS DADOS

O ambiente escolhido para o desenvolvimento do projeto foi o Google Colab. Esse ambiente foi escolhido por ser fácil de utilizar e possuir um ambiente com GPU, sendo possível treinar as redes neurais em um tempo hábil.

Como o dataset está na plataforma Kaggle, para baixar as imagens no Google Colab foi necessário criar uma conta no Kaggle e criar um Token, sendo possível importar para esse novo ambiente de desenvolvimento.

As imagens do dataset estão organizadas em pastas de treino, teste e validação. Dentro dessas pastas contém mais três pastas contendo as imagens de raio-x de pessoas com COVID-19, outras doenças (NON-COVID) e sem nenhum problema (NORMAL).

Para ser possível trabalhar com as imagens no programa foi criado uma função denominada *leImagens* que tem como objetivo retornar um conjunto de imagens em escala de cinza, juntamente com os seus rótulos. Essa função recebe quatro parâmetros: *wildcards* com os caminhos das imagens, os rótulos de de cada conjunto de imagens da lista de *wildcards* (“1” caso tenha COVID-19 e “0” caso não tenha COVID-19) e o número de linhas e colunas que o usuário deseja que as suas imagens tenham.

O tamanho original das imagens é de 256x256 pixels, mas será realizado um resize para 224x224 pixels.

Para ser possível a observação das imagens que serão testadas no modelo, juntamente com os seus rótulos, foi criado uma função *leImagensRGB* com o objetivo de retornar as imagens de raio-x na escala RGB e ficar melhor de visualizá-las.

Na Figura 01 é possível observar um lote de imagens de raio-x com os rótulos de COVID-19 e Non-COVID.

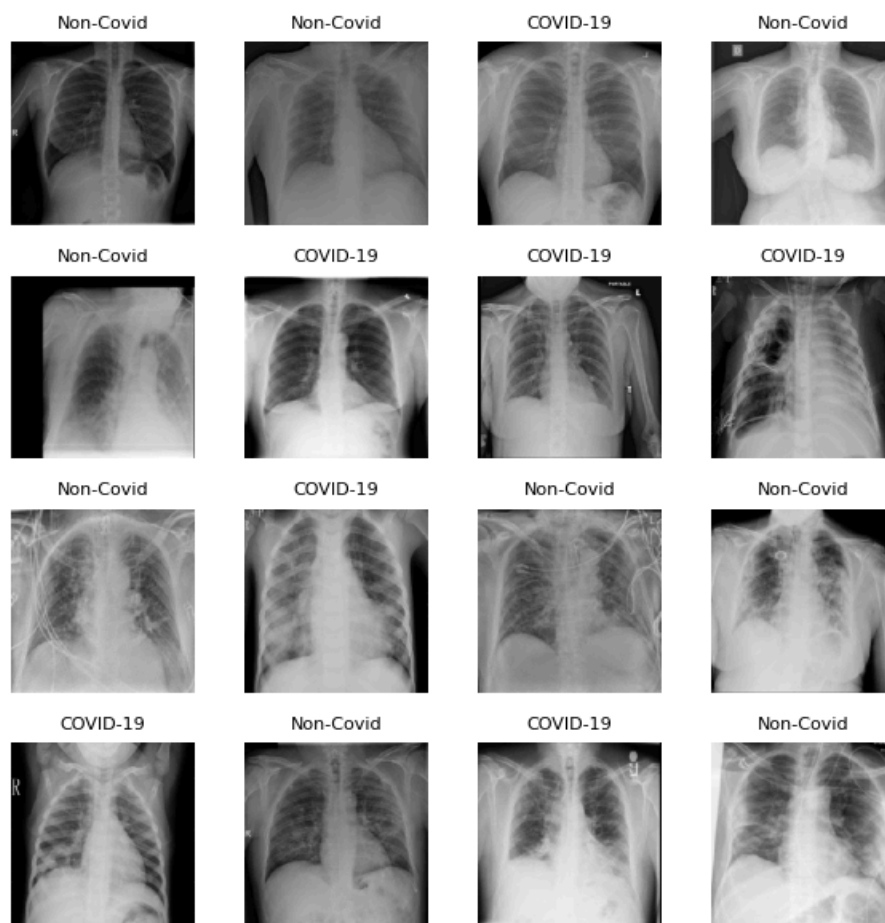


Figura 1 - Lote de raio-x com os rótulos de COVID-19 e NON-COVID

3. CLASSIFICAÇÃO - REDE NEURAL TREINADA DO ZERO

Para a classificação das imagens criando uma rede neural do zero, foi criada uma rede sem se basear em uma rede já existente. Os pesos da rede foram inicializados randomicamente. A rede foi criada utilizando a API funcional do Keras.

A rede contém quatro camadas convolucionais, com 50, 60, 70 e 80 neurônios, respectivamente, foi utilizado um strides de 2 e função de ativação ReLu, entre cada camada tem uma camada de dropout para reduzir a chance de ocorrer overfitting. Além disso, depois das camadas convolucionais e dropout de 0,25, existe uma camada flatten e duas camadas densas. Uma camada densa contém 1024 neurônios, após ela existe uma outra camada de dropout de 0,25 e por fim a última

camada densa com um único neurônio para classificar as imagens. Na Figura 2 é possível observar a rede criada.

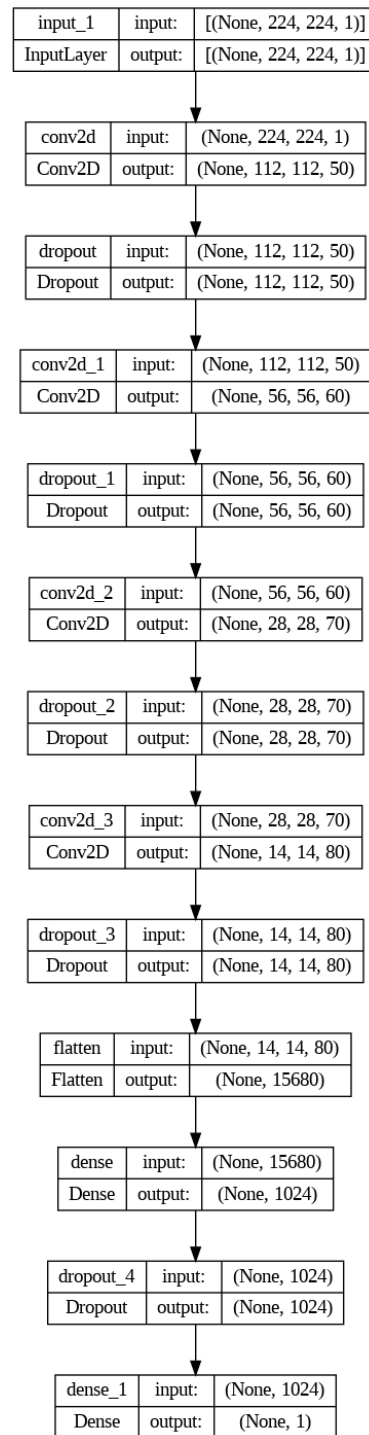


Figura 2 - Rede neural criada para classificação de raio-x

3.1. Treinamento

O treinamento foi realizado com 18 épocas, pois foi um valor que obteve um bom resultado. A acurácia e o loss durante as épocas no treinamento pode ser visto na Figura 3 e Figura 4.

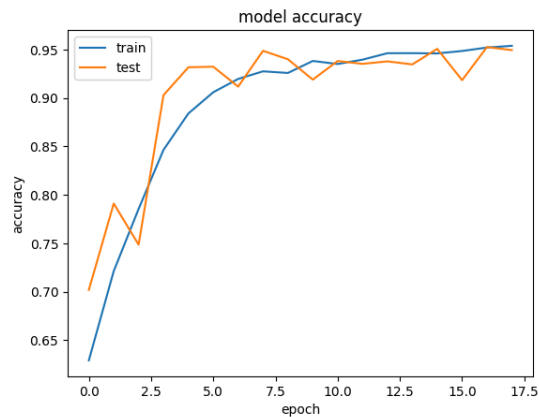


Figura 3 - Acurácia modelo sem transfer learning

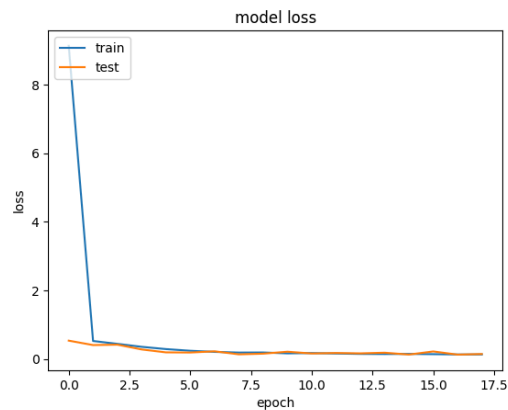


Figura 4 - Loss modelo sem transfer learning

Cada época levou um tempo médio de 36 segundos, levando 675 segundos para o treinamento ser concluído.

3.2 Teste e Curva ROC

Após o treinamento, foi realizada a predição do modelo com os dados de teste. A acurácia obtida foi de 95,01%. A sensibilidade do teste foi 43,8% e a especificidade de 56,2%.

Utilizando a biblioteca metrics do sklearn foi obtido um AUC de 0,989. Através dos dados da AUC, foi possível obter o ponto EER. O EER TPR foi de 0,954 e EER FPR foi de 0,0455. Com isso, foi possível plotar a curva ROC juntamente com o ponto EER, visto na Figura 5.

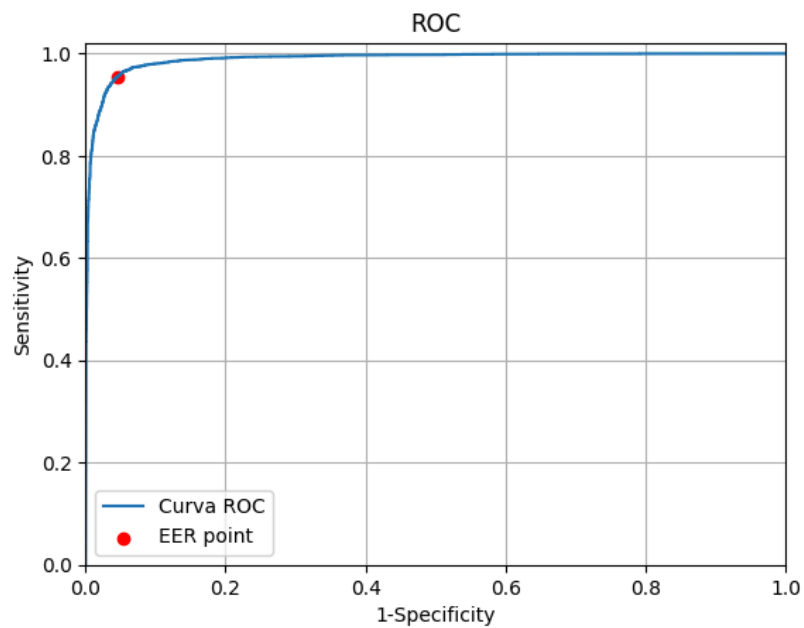


Figura 5 - Curva ROC modelo sem transfer learning

4. CLASSIFICAÇÃO - REDE NEURAL COM TRANSFER LEARNING

O modelo com transfer learning treinado no ImageNet foi o ResNet50. A rede ResNet50 foi treinada com imagens de 224x224 pixels, como já redimensionamos as imagens para esse formato não será necessário alterar o tamanho das delas. A estrutura da rede ResNet50 pode ser vista na Figura 6.

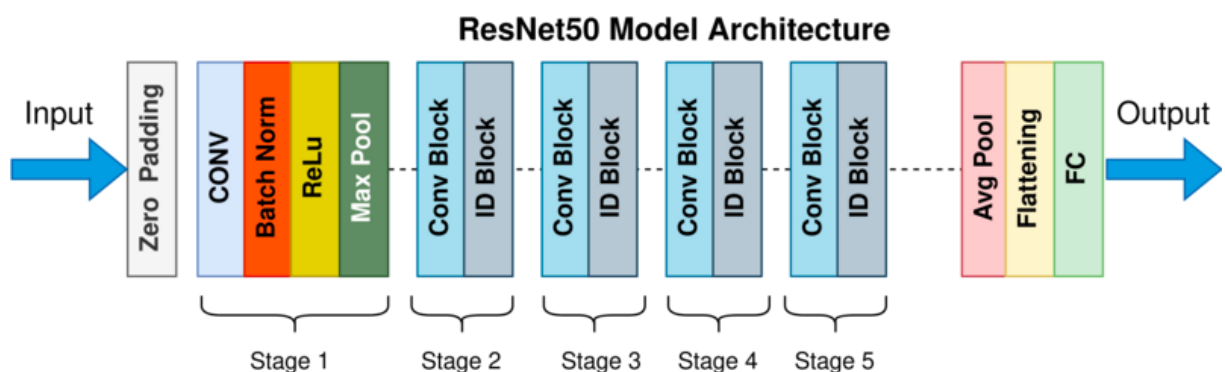


Figura 6 - Arquitetura do modelo ResNet50

Fonte: <https://commons.wikimedia.org/wiki/File:ResNet50.png>

Na saída da ResNet50 foi aplicado uma camada de global average pooling, flatten e uma camada densa com um neurônio e função de ativação sigmoid para classificar a imagem em COVID-19 (“1”) ou Sem COVID-19 (“0”). O treinamento foi realizado em duas partes, sendo a primeira parte congelando as camadas-base da rede ResNet50 e a segunda parte realizando o Fine Tuning, descongelando essas camadas-base.

4.1 Treinamento congelando as camadas-base

O treinamento desta rede foi realizado em duas partes. Para a primeira parte foi congelado as camadas-base da rede. Essa primeira etapa foi realizada em 15 épocas. Na Figura 7 e Figura 8 é possível observar a acurácia e o loss durante cada época. Além disso, foi realizado o otimizador Adam com o learning rate de 1×10^{-4} .

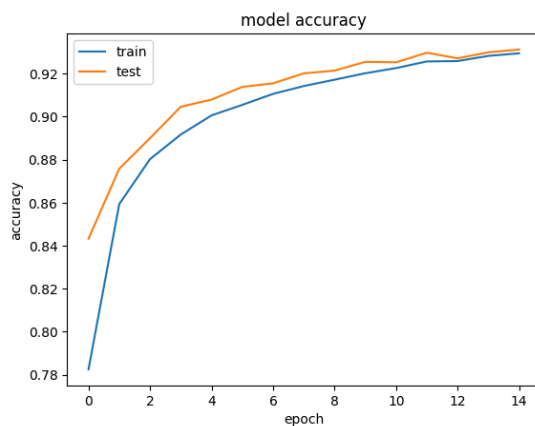


Figura 7 - Acurácia primeiro treinamento

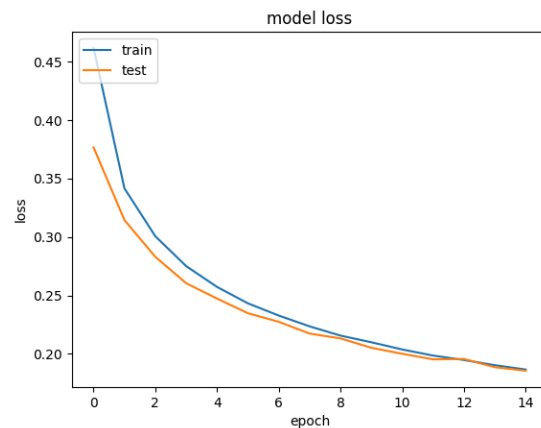


Figura 8 - Loss primeiro treinamento

Após esse primeiro treinamento foi realizado um teste no conjunto de teste, obtendo uma acurácia de 92,65%.

Para concluir essa parte do treinamento levou 1251 segundos, levando aproximadamente 89 segundos para cada época.

4.2 Fine Tuning

A segunda parte do treinamento foi realizado o Fine Tuning, descongelando as camadas-base da ResNet50 e treinando durante 8 épocas. Para realizar o fine tuning, foi colocado no otimizador de Adam um learning rate menor 1×10^{-5} . Na Figura 9 e Figura 10 é possível observar a acurácia e o loss durante cada época de treinamento.

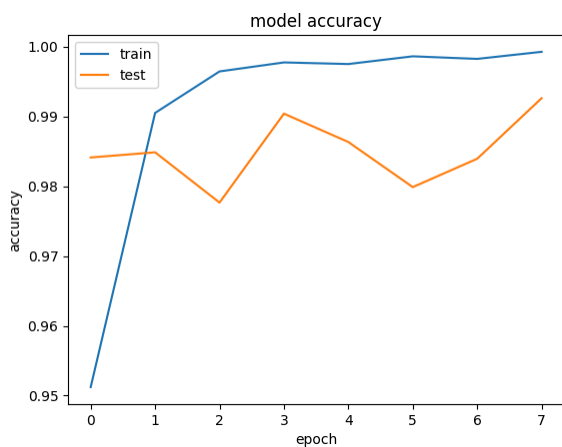


Figura 9 - Acurácia Fine Tuning

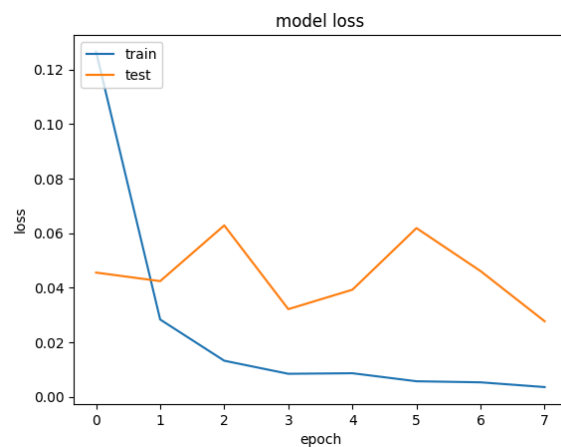


Figura 10 - Loss Fine Tuning

Para realizar o treinamento dessa parte, em média cada época levou quase 260 segundos, sendo um total de 2077 segundos só para realizar o treinamento de 8 épocas.

4.3 Teste e curva ROC

Por fim, foi realizada a predição do modelo treinado com os dados de teste. A acurácia obtida foi de 99,1%. Além disso, a sensibilidade foi de 41,11% e a especificidade de 58,89%. O valor do AUC foi de 0,9993.

O eer tpr foi de 0,99 e eer fpr foi de 0,0093. Plotando a curva ROC juntamente com o ponto EER, foi obtida a Figura 11.

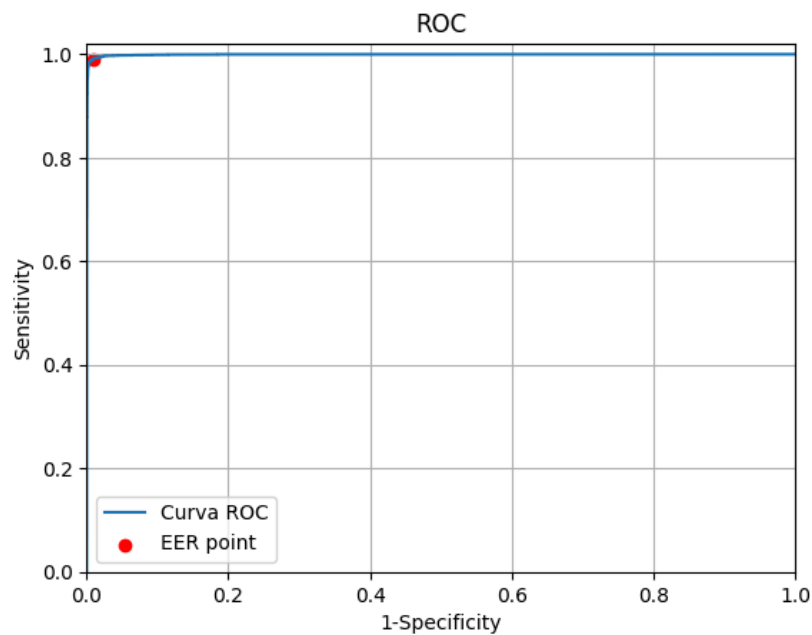


Figura 11 - Curva ROC modelo com ResNet50

5. CONCLUSÃO

Portanto, é possível concluir que o melhor modelo foi o modelo com transfer learning. O resultado obtido foi muito bom, com uma alta acurácia (99,1%) e um AUC de 0,9993 muito próximo de 1.

Contudo, o tempo de treinamento do modelo com transfer learning é bem superior ao modelo sem. Além disso, o resultado foi satisfatório do modelo sem transfer learning, chegando em uma acurácia maior que 95% com apenas 18 épocas e um tempo de treinamento muito menor comparado ao modelo com transfer learning.

REFERÊNCIAS

THUSEETHAN, Selvarajah; WILMALASOORIYA, Chathrie; VASANTHAPRIYAN, Shanmuganathan. **Deep COVID-19 Recognition using Chest X-ray Images: A Comparative Analysis**. International Conference on Artificial Intelligence, 2021.

ANEXOS

Link para o notebook com o modelo sem transfer learning:
<https://colab.research.google.com/drive/15IQ5592ZZV9a7UwizLM7M2ijTgQsgsGU?usp=sharing>

Link para o notebook com o modelo de RESNet50:
https://colab.research.google.com/drive/128aG_ap5MucIGhXXBTbIVvxVB8e2Yzyn?usp=sharing