

Uso de RNC para o diagnóstico de COVID-19 por imagens de tomografia computadorizada do pulmão

Débora Daniela Wendland Amorim
Departamento de Ciências Exatas e Sociais
Aplicadas
Universidade Federal de Ciências da Saúde de
Porto Alegre
Porto Alegre, Brasil
debora@ufcspa.edu.br

Abstract—This work aims to describe the development of a convolutional neural network to be implemented in the COVID-19 diagnosis support using computed tomography images of infected patients.

Keywords—COVID-19, diagnosis, computed tomography, machine learning, convolutional neural network

Resumo—Este trabalho visa relatar a construção de uma rede neural convolucional para ser implementada no auxílio ao diagnóstico de COVID-19 utilizando imagens de tomografia computadorizada de pulmão de pacientes infectados.

Palavras-chave—COVID-19, diagnóstico, tomografia computadorizada, aprendizado de máquina, rede neural convolucional

I. INTRODUÇÃO

A COVID-19, causada pelo vírus Sars-CoV-2, é uma doença infecciosa que já causou a morte de mais de 250 mil pessoas e infectou mais de 10 milhões de pessoas somente no Brasil. Por mais que tenhamos estes impressionantes números de contaminados e mortes, a falta e desperdício de testes ainda é uma realidade no Brasil. Em períodos onde a falta de teste se torna grande seja pela alta demanda ou pela baixa disponibilidade, as imagens de tomografia computadorizada podem ser usadas para o auxílio no diagnóstico de COVID-19. As imagens auxiliam no diagnóstico pelo fato de ser possível distinguir uma pneumonia viral de outras formas de pneumonia e esta pneumonia viral é provavelmente causada por COVID-19, o que já auxilia não só no diagnóstico, mas também nas medidas de cuidado com o paciente.

II. CONJUNTO DE DADOS

O conjunto de dados escolhido [1] para realização deste trabalho é formado por imagens de tomografia computadorizada do pulmão extraídas de artigos relacionados a COVID-19 dos seguintes lugares: medRxiv, bioRxiv, NEJM, JAMA, Lancet, etc. As imagens estão nos formatos .png e .jpg.

O conjunto de dados disponibilizado no github já conta com o conjunto de imagens pré-processadas e segmentadas nas três etapas distintas de treinamento, validação e testes, totalizando 746 imagens. A divisão pode ser visualizada na tabela 1.

TABELA I. DIVISÃO DO DATASET EM SUAS ETAPAS

Etapa	Classes		
	Total	COVID	NonCOVID
Treinamento	425	191	234
Validação	118	60	178
Teste	203	98	105

O dataset é dividido em duas classes: COVID e NonCOVID, onde exemplos das imagens das classes podem ser conferidas na fig 1 e 2.

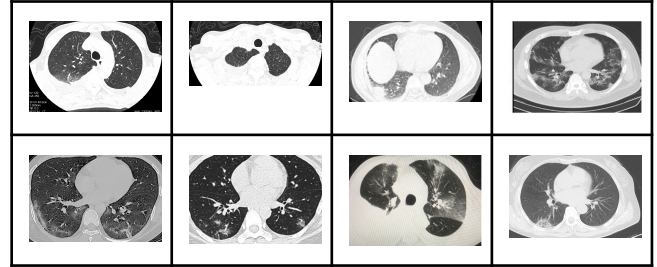


Fig. 1. Exemplos de imagens de CT positivas para COVID, classe: COVID.

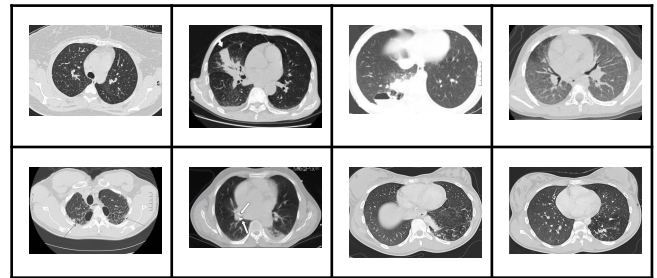


Fig. 2. Exemplos de imagens de CT negativas para COVID, classe: NonCOVID.

III. INTERFACE DE CARREGAMENTO DE DADOS

Para o carregamento dos dados, inicialmente é baixado os arquivos de COVID e NonCOVID do repositório no github, após elas são extraídas. Da mesma forma, é baixado os arquivos que constam a informação da divisão das imagens em treinamento, validação e teste. Após as imagens são separadas em três pastas diferentes, respectivamente, com as subpastas identificando as classes. Esta informação está disponível no arquivo *data.json*.

É utilizada a classe *ImageDataGenerator* para realizar o *rescale* das imagens que antes possuíam os pixels era no intervalo de 0 a 255, agora estarão no intervalo de 0 a 1 que é preferido para modelos de redes neurais. Desta classe, é utilizada a função *flow_from_directory* para importar as imagens de validação e treinamento. Nessa importação, o target da imagem é dimensionado para 227x227 para se tornar compatível com o modelo, o modo de classes é categórico e o tamanho do batch foi mantido 32 por default, como pode ser observado na fig. 3.

```
Batch Size for Input Image : (32, 227, 227, 3)
Batch Size for Output Image : (32, 2)
Image Size of first image : (227, 227, 3)
Output of first image : (2,)
```

Fig. 3. Tamanho do batch para imagem de entrada e saída, bem como tamanho da imagem de entrada e saída.

IV. ARQUITETURA DA REDE NEURAL CONVOLUCIONAL

O modelo de arquitetura escolhido é baseado em AlexNet [3]. O modelo utiliza a função ReLU como função de ativação inicial e é concebida inicialmente com 8 camadas, sendo 5 camadas convolucionais e 3 camadas densas. Cada camada convolucional é seguido de uma camada de max pooling e também utiliza dropout nas duas primeiras camadas totalmente conectadas para evitar o overfitting. Também utiliza como algoritmo de otimização o gradiente descendente estocástico. Maiores informações sobre o modelo podem ser conferidas na fig. 4.

Model: "AlexNet"		
Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 55, 55, 96)	34944
max_pooling2d (MaxPooling2D)	(None, 27, 27, 96)	0
conv2d_1 (Conv2D)	(None, 27, 27, 256)	614656
max_pooling2d_1 (MaxPooling2D)	(None, 13, 13, 256)	0
conv2d_2 (Conv2D)	(None, 13, 13, 384)	885120
conv2d_3 (Conv2D)	(None, 13, 13, 384)	1327488
conv2d_4 (Conv2D)	(None, 13, 13, 256)	884992
max_pooling2d_2 (MaxPooling2D)	(None, 6, 6, 256)	0
Flatten (Flatten)	(None, 9216)	0
dense (Dense)	(None, 4096)	37752832
dropout (Dropout)	(None, 4096)	0
dense_1 (Dense)	(None, 4096)	16781312
dropout_1 (Dropout)	(None, 4096)	0
dense_2 (Dense)	(None, 1000)	4097000
dense_3 (Dense)	(None, 2)	2002
Total params: 62,380,346		
Trainable params: 62,380,346		
Non-trainable params: 0		

Fig. 4. Modelo AlexNet [4]

Para otimização dos resultados foram realizadas as seguintes modificações no modelo AlexNet (fig. 5):

- Diminuição das camadas convolucionais de 5 para 2, com 32 e 64 filtros, respectivamente.
- Diminuição das camadas densas de 3 para 2, com 128 e 2 filtros respectivamente.
- Remoção do Dropout entre as camadas totalmente conectadas.
- Na primeira camada densa foi alterada a função de ativação de ReLU para Sigmóide, devido ao fato da sigmóide ser melhor para a classificação binária, que é a que estamos realizando entre duas classes.

Model: "AlexNet"		
Layer (type)	Output Shape	Param #
conv2d_71 (Conv2D)	(None, 55, 55, 32)	11648
max_pooling2d_67 (MaxPooling2D)	(None, 27, 27, 32)	0
conv2d_72 (Conv2D)	(None, 27, 27, 64)	18496
max_pooling2d_68 (MaxPooling2D)	(None, 13, 13, 64)	0
flatten_28 (Flatten)	(None, 10816)	0
dense_70 (Dense)	(None, 128)	1384576
dense_71 (Dense)	(None, 2)	258
Total params: 1,414,978		
Trainable params: 1,414,978		
Non-trainable params: 0		

Fig. 5. Modelo adaptado de AlexNet

V. METODOLOGIA DE TREINAMENTO E OBTENÇÃO DO MODELO

Para o treinamento da RNC foi utilizado um otimizador Adam sendo baseado no SGD, porém traz o benefício de ser mais rápido em seu processamento. Para a função de perda, foi utilizada `categorical_crossentropy` que estava por padrão no AlexNet de modelo. E para a métrica, foi mantido a acurácia como método desejado.

Para configurar o modelo de AlexNet, foram utilizados os seguintes hiperparâmetros:

- `input_shape`: formato do input referente as imagens de 227x227x3;
- `num_classes`: número de classes para a categorização de 2 (COVID e NonCOVID).

Para realizar o treino do modelo, foram passados os datasets de treinamento e validação, bem como o modelo desejado e o número de épocas que foram utilizadas 30. Não foram utilizados um número maior de épocas por se avaliar que elas não incrementaram a acurácia do modelo que foi adaptado.

Após treinar e adaptar os hiperparâmetros, podemos analisar os gráficos de perda e acurácia, fig 6 e 7, respectivamente. Podemos perceber que para a função de perda, temos um desempenho ruim do dataset de validação, tendo valores acima de 1, fato que não é replicado com o dataset de treino que alcançou boas medidas para a perda, chegando a valores bons de 0.037. Similarmente à função de perda, no cálculo de acurácia, podemos observar o mesmo padrão, com um valor excelente de acurácia com o dataset de treino, mas que não é replicado com o dataset de validação, chegando a 69.5% de acurácia.

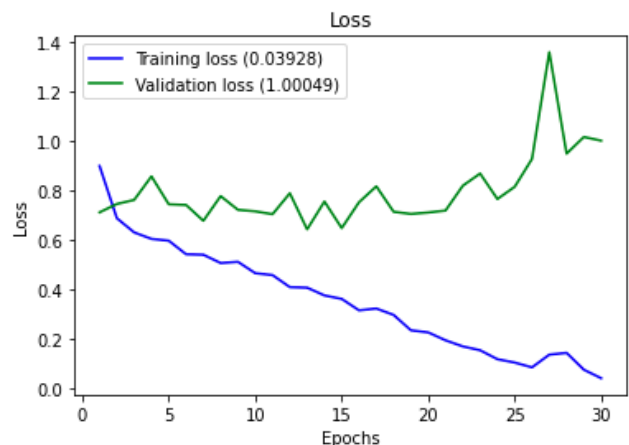


Fig. 6. Modelo adaptado de AlexNet

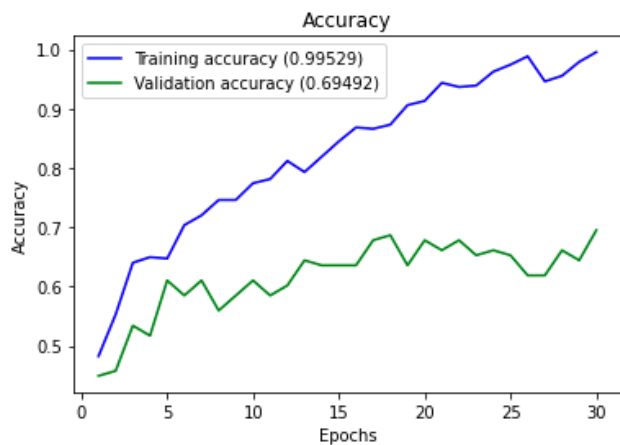


Fig. 7. Modelo adaptado de AlexNet

VI. CLASSIFICAÇÃO DOS DADOS

Com a predição de resultados utilizando o dataset de teste, conseguimos valores similares aos de validação, com os resultados sendo descritos na tabela II e III. A percentagem de acurácia obtida foi de 65.52%, onde se totalizam as percentagens de verdadeiros-positivos e verdadeiros-falsos. Da mesma forma, conseguimos totalizar 34.48% de erro na classificação dos dados. O valor de acurácia obtido é inferior, mas próximo, do valor de acurácia de validação, 69,5%.

TABELA II. RESULTADO DA CLASSIFICAÇÃO DOS DADOS DE TESTE

Classe Original	Resultado		
	COVID	NonCOVID	Total
COVID	64	34	98
NonCOVID	36	69	105
Total	100	103	203

TABELA III. COMPARAÇÃO ENTRE RESULTADOS FALSOS E VERDADEIROS, EM PORCENTAGEM

Classe Original	Resultado		
	Positivo	Negativo	Total
Positivo	31.53%	16.75%	48.28%
Negativo	17.73%	33.99%	51.72%
Total	49.26%	50.74%	100%

VII. RESULTADOS

Analisando os resultados obtidos do dataset de teste, podemos notar que não há uma diferença significativa na acurácia para as diferentes classes, onde a classe NonCOVID obteve uma taxa de falsos-positivos de 34.26% e na classe COVID, 34.69% de falsos-negativos.

Com isso, podemos concluir que é possível obter resultados verdadeiros em 65.52% das imagens testadas. Esse valor ainda pode ser melhorado aprimorando o modelo utilizado, bem como alterando os pacotes de treinamento, teste e validação que foram mantidos os originais utilizados no projeto-base.

REFERENCES

- [1] J. Zhao, Y. Zhang, X. He, P. Xie. "COVID-19 Lung CT Scans: COVID-CT-Dataset: A CT Scan Dataset about COVID-19," Disponível em: <https://www.kaggle.com/luisblanche/covidct>. Acesso em: 20 mar 2021
- [2] J. Zhao, Y. Zhang, X. He, P. Xie. "COVID-CT," Disponível em: <https://github.com/UCSD-AI4H/COVID-CT>. Acesso em: 20 mar 2021
- [3] A. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012.
- [4] A. Nejad. "Convolutional Neural Network Champions — Part 2: AlexNet (TensorFlow 2.x)" Disponível em: <https://towardsdatascience.com/convolutional-neural-network-champions-part-2-alexnet-tensorflow-2-x-de7e0076f3ff>. Acesso em: 10 mar 2021
- [5] Li L, Qin L, Xu Z, et al. Using Artificial Intelligence to Detect COVID-19 and Community-acquired Pneumonia Based on Pulmonary CT: Evaluation of the Diagnostic Accuracy. Radiology. 2020;296(2):E65-E71. doi:10.1148/radiol.202000905
- [6] Lu Huang, Rui Han, Tao Ai, Pengxin Yu, Han Kang, Qian Tao, and Liming Xia. Serial quantitative chest ct assessment of covid-19: deep-learning approach. Radiology: Cardiothoracic Imaging, 2(2):e200075, 2020.