

Detecção de COVID-19 por Imagens de Raio-X Usando Redes Neurais Convolucionais

1st Rafael Bortolozo

Instituto Federal de Educação Ciência e Tecnologia Catarinense, Videira, Brasil

rafaelbortolozo@outlook.com

Resumo—O presente trabalho tem como objetivo a apresentação de um modelo de inteligência artificial capaz de classificar imagens de raio-X, informando como resultado se o paciente possui ou não a COVID-19. Para isso, será utilizado uma rede neural convolucional aliada a uma base de dados pública contendo tanto imagens de pulmões de pacientes saudáveis quanto daqueles que foram contaminados pela doença. O modelo apresentou uma acurácia de 81,75% com 2000 imagens e 50 épocas.

Index Terms—Inteligência Artificial, Aprendizado Profundo, COVID-19, Rede Neural Convolucional, Raio-X

I. INTRODUÇÃO

Em 2020, uma nova pandemia surgiu com a COVID-19 ou novo coronavírus, do-ença pertencente à classe das Síndromes Respiratórias Agudas Grave (SRAG). O coronavírus foi identificado pela primeira vez em 1937 (INTERMÉDICA, 2020), sendo a variante SARS-COV-2 a responsável pela pandemia do “novo coronavírus”, que foi inicialmente identificada na China, na cidade de Wuhan, no final de 2019 (KIM; GANDHI, 2022).

Segundo o site da secretaria de estado de saúde de Minas Gerais (GERAIS, 2020), as pessoas contaminadas pela COVID-19 podem ou não apresentar sintomas, podendo ser desde casos leves até os mais graves que podem levar a vítima à óbito em questão de horas, então neste caso um diagnóstico rápido é de extrema importância.

A necessidade de testagem rápida em massa da população, o alto custo orçamentário no enfrentamento da pandemia, as milhões de vidas prejudicadas e a busca por conhecimentos relacionados ao pós-pandemia servem como estímulo para o desenvolvimento de novas ferramentas tecnológicas de diagnóstico.

Segundo Vandenberg Delphine Martiny e Kozlakidis (2020), não se sabe qual será o potencial negativo do vírus no futuro, quanto antes as ferramentas entrarem em funcionamento melhor será o combate. Essas ferramentas são geralmente desenvolvidas usando alguma abordagem de aprendizado de máquina (AM), ou *Machine learning* no idioma inglês, pois essa técnica de Inteligência Artificial (IA) consegue aprender com um conjunto de dados de treinamento e passar a analisar outros dados com base no seu aprendizado.

Segundo Carleo et al. (2019), o AM se tornou uma técnica bastante utilizada para resolver diversos tipos de problemas na atual década, tendo um grande potencial na análise de dados e detecção de características por meio do reconhecimento de

padrões. Estes modelos têm obtido resultados impressionantes na detecção de padrões superando a capacidade humana em diversos domínios de aplicação.

Entretanto, neste trabalho a intenção é que o desenvolvimento do aprendizado de máquina seja feito através de imagens radiológicas, especialmente imagens de Raio-X (RX), já que existem diversos datasets públicos contendo milhares de imagens RX de pacientes saudáveis e contaminados, então trata-se de um trabalho de visão computacional.

No mundo da visão computacional, é usado geralmente uma abordagem chamada Rede Neural Convolucional (RNC), cujo tema receberá maior enfoque no próximo capítulo. As RNC pertencem a área de Aprendizado Profundo (AP), ou *Deep Learning* no idioma inglês, esta sendo uma subárea do AM.

O AP se diferencia do AM principalmente na forma como é implementado o extrator de características, uma das etapas para a detecção de padrões. No AP o extrator de características é construído e moldado pela própria RNA e não pelo desenvolvedor, assim aumentando consideravelmente a capacidade de detecção da rede, não se limitando pelo raciocínio humano (LECUN; HINTON, 2015).

No presente artigo, busca-se implementar uma RNC para detecção de pacientes com COVID-19, bem como sua metodologia e os resultados obtidos com o modelo proposto.

II. REFERENCIAL TEÓRICO

O presente capítulo irá apresentar uma breve fundamentação teórica sobre AP e especialmente sobre RNC e suas camadas.

A. Aprendizado Profundo

O AP é uma subárea do Machine Learning muito popular pela sua colaboração na resolução de diversos problemas da vida real devido às suas várias camadas de abstração, especialmente quando o problema envolve a análise de imagens, sendo as redes neurais convolucionais o paradigma mais utilizado para esse objetivo (SHEN; WU; SUK, 2017; KER et al., 2018).

Um importante detalhe do aprendizado profundo é que geralmente é exigido uma quantidade de dados bem maior do que em relação as redes convencionais, isso porque em uma rede AP existem muito mais camadas ocultas, na escala de dezenas ou centenas, cada uma executando uma tarefa diferente, aumentando consideravelmente a complexidade computacional. Além disso, tem a questão do extrator de características, que pode ser visualizado na figura 1, que não é definido pelo desenvolvedor e sim pela rede, exigindo

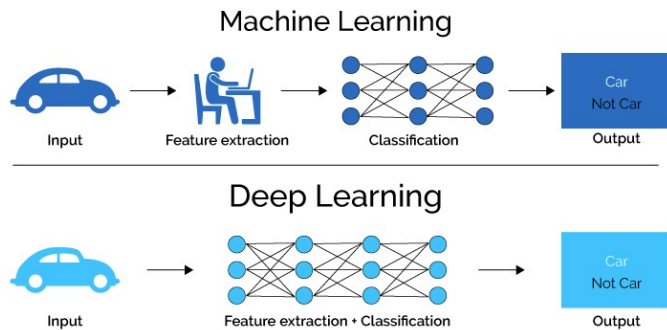


Figura 1. Acima, o funcionamento de um AM onde o desenvolvedor é responsável pela extração de características e abaixo o funcionamento de um AP onde a extração de características é integrado à rede neural.

mais dados. Uma AP com poucos dados pode resultar em uma sensibilidade muito alta a mudanças de características, porém há algumas técnicas que podem mitigar esse problema, como o *Data Augmentation* e Transferência de conhecimento (HOARFROST A. APTEKMANN; BROMBERG, 2022).

Resumidamente, uma AP é uma rede neural com muitas camadas de neurônios, que é alimentada pela camada de entrada, esta recebendo os dados brutos. Cada camada faz uma abstração dos dados para a próxima camada, que por fim retorna uma classe na ultima camada, a camada de saída (LECUN; HINTON, 2015).

B. Rede Neural Convolucional

A RNC é um tipo de rede AP que obteve notável reconhecimento pelo seu sucesso em tarefas de visão computacional graças a convolução, o principal método da RNC, desenvolvido por LeCun em 1995 (LECUN; BENGIO et al., 1995). Na área da medicina, Yasaka Hiroyuki Akai e Abe (2018) apresenta uma noção de como as RNCs possuem um grande potencial na detecção e classificação de lesões e também em outras tarefas envolvendo imagens radiológicas.

O principal diferencial dela em relação a outros métodos de visão computacional é a forma como a imagem é tratada na entrada, que ao invés de passar todos os valores de pixel da imagem como entrada na rede neural, uma RNC olha para diversas partes da imagem e tenta identificar o que os pixels estão querendo representar em cada parte, dessa forma a rede passa a classificar imagens muito distintas (que pertencem a mesma classe) independente da rotação, translação, escala e transformações das características (AREL et al., 2010).

A RNC é dividida em duas fases, a primeira sendo a extração de características e a segunda a classificação com RNA, que recebe como entrada o que foi interpretado na fase anterior para identificar o padrão pretendido (O'MAHONY et al., 2019). A fase de extração de características é composta entre camadas de convolução e *pooling*. A estrutura geral de uma RNC pode ser visualizado na figura 2.

Na camada de convolução é aplicado diversos filtros, conhecidos como *kernels* no idioma inglês, que são matrizes geralmente 3x3 que são multiplicadas por toda a matriz da imagem, criando cópias da imagem que apresentam as características

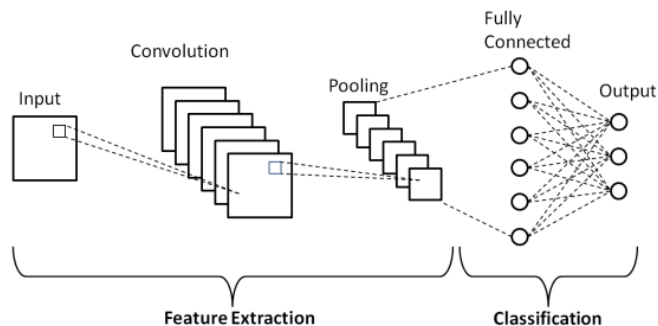


Figura 2. Visualização da estrutura de uma RNC apresentando as duas fases, as camadas de convolução e pooling e a RNA para classificação.

priorizadas pelo filtro e que serão usadas como parâmetro nas camadas seguintes. Geralmente são usadas para destacar linhas verticais, horizontais, círculos, quadrados e entre outras características (O'MAHONY et al., 2019). Para um melhor entendimento, pode-se fazer uma analogia ao peneirar uma farinha, onde a farinha é aplicada à peneira (que prioriza grãos finos), gerando um novo produto onde foi destacado apenas as partículas compatíveis com a peneira (partículas menores).

Na camada de *pooling* busca-se resumir as características de uma entrada, diminuindo assim a resolução da entrada e aumentando a velocidade de treinamento sem que haja a perda de características essenciais para o treinamento. Essa camada geralmente é aplicada após a camada de convolução. Para o seu funcionamento, é necessário informar o tamanho da matriz de *pooling* e a regra de ativação. O *pooling* funciona de maneira semelhante à convolução, com a diferença que o passo do filtro sobre a imagem (conhecido como *stride*) é definido pelo tamanho da matriz de *pooling* e o valor gerado é definido por uma regra de ativação, que pode ser de máximo onde apenas o maior elemento da matriz é selecionado, média onde será feita a média de todos os valores ou mínima onde apenas o menor elemento da matriz é selecionado. Na prática, se optar pelo *pooling* máximo, uma matriz 3x3 será resumida com o maior valor (O'MAHONY et al., 2019).

Ao final da fase de extração de características, já na etapa de classificação, é feita uma operação chamada *Flatten*, que consiste em pegar o resultado em matriz obtido pelo encadeamento de camadas convolucionais e de *pooling* e transformá-lo em vetor, desta forma é possível passar os dados para uma RNA (O'MAHONY et al., 2019).

III. METODOLOGIA

O intuito do trabalho é criar uma Rede Neural Convolutiva que detecta COVID-19 em imagens de Raio-X. Primeiramente foi preciso achar um banco de imagens público e gratuito contendo imagens RX de pacientes positivados e negativados pela COVID-19, onde foi optado por um dataset chamado "COVID-19 Radiography Database", disponibilizado na plataforma "Kaggle". Na data de escrito deste trabalho, o dataset contém 3.616 imagens RX de positivados e 10.206 em estado normal.

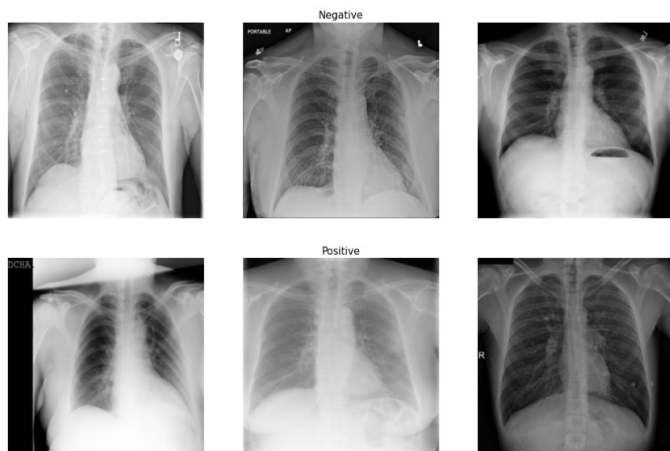


Figura 3. Exemplos de seis imagens aleatórias, três de cada classe, que foram devidamente mapeadas pelo algoritmo.

O algoritmo foi desenvolvido na linguagem de programação *Python*, dentro da plataforma *Colaboratory* (também chamado de Colab) da *Google*. Há um porém em relação ao dataset, que será utilizado apenas 1000 imagens aleatórias de cada classe, totalizando 2000 imagens do máximo total de 13.822, isso se dá ao fato que o Google Colab demora muito tempo tanto para baixar todas as imagens do dataset quanto no treinamento da rede, mesmo estando o ambiente de execução configurado com aceleração de hardware.

A implementação do algoritmo é relativamente simples. Foi usado duas tecnologias no desenvolvimento da RNC, o *TensorFlow* e o *Keras* e aliado ao ambiente de desenvolvimento, facilita muito o uso dessas ferramentas por já estarem instaladas nos servidores da *Google*.

Com o dataset importado, o primeiro passo foi ler e mapear as imagens em casos positivos e negativos, isso foi feito usando o nome de cada pasta em que a imagem está inserida. A figura 3 apresenta algumas imagens RX aleatórias mapeadas.

Antes de dividir os dados as imagens para o treinamento, foram feitas algumas mudanças sob as imagens:

- Imagens foram redimensionadas, passando da resolução padrão 299x299 para 70x70 no intuito de aumentar a velocidade do modelo;
- Imagens se transformaram em vetor de informações usando a biblioteca *numpy*, incluindo a intensidade dos pixels [0-255] de cada cor do espectro RGB e o resultado do paciente (0 ou 1);
- Valores foram normalizados, isso colabora para um bom treinamento do modelo, além de ajudar a evitar um *Overfitting*;

Com isso pronto, os dados foram divididos entre amostra para treinamento e amostra para teste, 80% (1600 imagens) e 20% (400 imagens). Com a divisão feita, o modelo de RNC pode ser construído. Foi usado as seguintes camadas:

- Convolução com 128 filtros de tamanho 3x3;
- Pooling máximo com filtro de tamanho 2x2;
- Dropout de 30%;

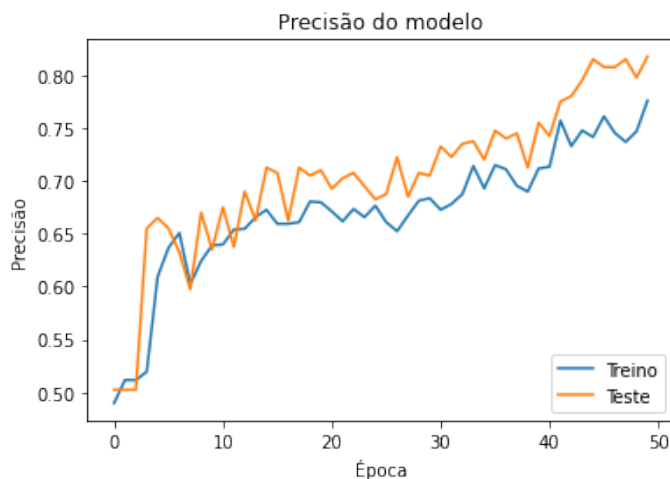


Figura 4. Precisão crescente do modelo em cada época para as imagens de treinamento e teste.

- Convolução com 64 filtros de tamanho 3x3;
- Pooling máximo com filtro de tamanho 2x2;
- Dropout de 50%;
- Convolução com 64 filtros de tamanho 3x3;
- Flatten;
- Camada densa de 16 neurônios;
- Dropout de 20%;
- Camada com duas saídas.

Com tudo pronto, agora é possível iniciar o treinamento do modelo chamando a função *model.fit()*. O treinamento foi realizado com apenas 50 épocas devido a limitações de hardware.

IV. RESULTADOS

Por haver falta de poder computacional na execução do algoritmo, o resultado foi limitado, porém foi possível extrair algumas informações interessantes e afirmar que o objetivo do trabalho foi atingido.

Após o treinamento do modelo, foi obtido uma acurácia de 81,75% nos dados de teste, o que é considerado um valor bem atrativo diante do ambiente de execução em que a rede foi submetida.

Nas figuras 4 e 5 é possível visualizar o desempenho do modelo durante cada época, apresentando curvaturas consideradas normais e principalmente, sem sinais de *overfitting*.

Na figura 6 é apresentada a matriz de confusão para os dados de teste. Essa matriz mostra as frequências de classificação para cada classe do modelo, ajudando a certificar se o modelo classificou as imagens corretamente. O ideal é que os maiores frequências estejam nas casas onde os índices são iguais (diagonal principal), ou seja, as classificações do modelo coincidem com o que era previsto (verdadeiros positivos/negativos).

V. CONCLUSÃO

Com base nos resultados obtidos, mesmo com limitações de hardware, percebe-se que o modelo tem grande potencial

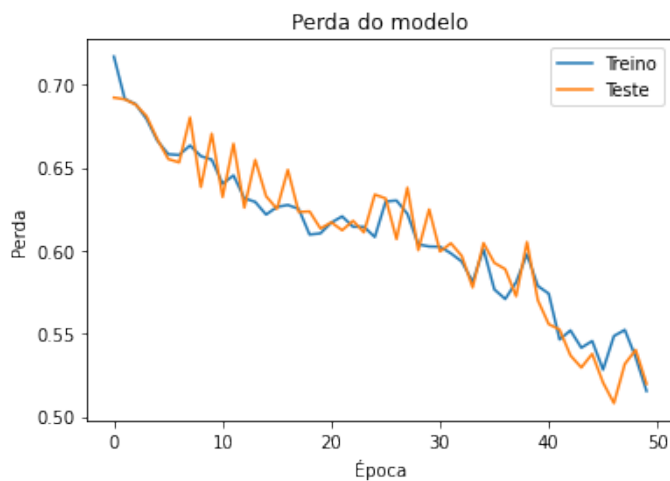


Figura 5. Perda decrescente do modelo em cada época para as imagens de treinamento e teste.

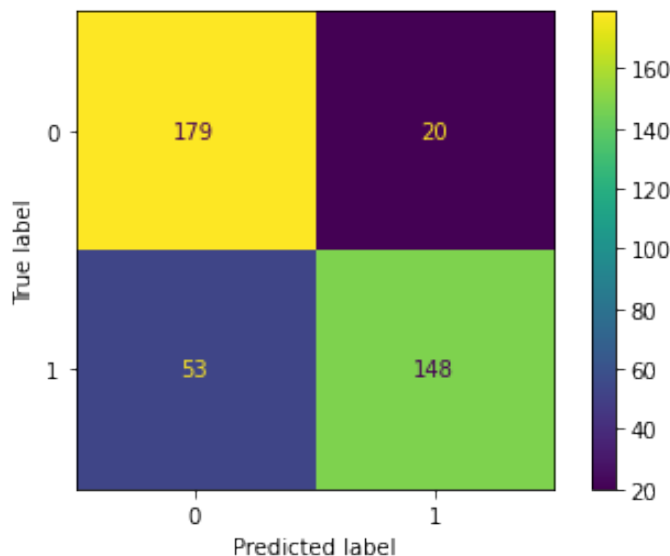


Figura 6. Matriz de confusão na classificação dos dados de teste.

para detectar a COVID-19 graças a grande facilidade de aprendizado com poucas imagens e épocas. Obviamente que o resultado do modelo proposto está longe de ser o melhor ou adequado para aplicar em um projeto final, sendo o ideal treinar a rede por muito mais tempo incrementando mais épocas, imagens no dataset, testando outras combinações de camadas e parâmetros, e principalmente utilizar um hardware condizente com a complexidade da rede.

Como sugestão para trabalhos futuros, é interessante aplicar técnicas como transferência de conhecimento usando algum modelo pré-treinado com imagens RX de COVID-19, usando o dataset como ajuste desse modelo e diminuindo o consumo de recursos computacionais. Além disso, é interessante aplicar a técnica de *Data Augmentation* para criar cópias de cada imagem, porém com algumas características alteradas como

brilho, contraste, rotação, saturação e entre outras. Isso ajuda a aumentar o volume do dataset, além de acostumar o modelo para classificar imagens com essas variâncias.

REFERÊNCIAS

- [1] INTERMÉDICA, NotreDame. Como surgiu o coronavírus e como afeta a população mundial. 2020. Disponível em: <https://www.gndi.com.br/saude/blog-da-saude/como-surgiu-o-coronavirus>.
- [2] KIM, Arthur Y; GANDHI, Rajesh T. COVID-19: Management in hospitalized adults. 2022.
- [3] GERAIS, Secretaria de estado de saúde de Minas. Você sabe o que é Síndrome Respiratória Aguda Grave? 2020. Disponível em: <https://coronavirus.saude.mg.gov.br/blog/75-o-que-e-sindrome-respiratoria-aguda-grave>.
- [4] VANDENBERG DELPHINE MARTINY, Olivier Rochas Alex van Belkum Olivier; KOZLAKIDIS, Zisis. Considerations for diagnostic COVID-19 tests. *Nat Rev Microbiol*, n. 19, p. 171–183, 2020. Disponível em: <https://doi.org/10.1038/s41579-020-00461-z>.
- [5] CARLEO, Giuseppe et al. Machine learning and the physical sciences. *Rev. Mod. Phys.*, American Physical Society, v. 91, p. 045002, Dec 2019. Disponível em: <https://link.aps.org/doi/10.1103/RevModPhys.91.045002>.
- [6] LECUN, Yoshua Bengio Yann; HINTON, Geoffrey. Deep learning. *Nature*, v. 521, n. 7553, p. 436–444, 2015.
- [7] SHEN, Dinggang; WU, Guorong; SUK, Heung-Il. Deep learning in medical image analysis. *Annual Review of Biomedical Engineering*, v. 19, n. 1, p. 221–248, 2017. PMID: 28301734. Disponível em: <https://doi.org/10.1146/annurev-bioeng-071516-044442>.
- [8] KER, Justin et al. Deep learning applications in medical image analysis. *IEEE Access*, v. 6, p.9375–9389, 2018.
- [9] HOARFROST A. APTEKMANN, G. Farfañuk A.; BROMBERG, Y. Deep learning of a bacterial and archaeal universal language of life enables transfer learning and illuminates microbial dark matter. *Nat Commun*, v. 13, n. 2606, 2022.
- [10] LECUN, Yann; BENGIO, Yoshua et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, Cambridge, MA USA, v. 3361, n. 10, p. 1995, 1995.
- [11] O'MAHONY, Niall et al. Deep learning vs. traditional computer vision. In: SPRINGER. *Science and information conference*. [S.l.], 2019. p. 128–144.
- [12] M.E.H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M.A. Kadir, Z.B. Mahbub, K.R. Islam, M.S. Khan, A. Iqbal, N. Al-Emadi, M.B.I. Reaz, M. T. Islam, "Can AI help in screening Viral and COVID-19 pneumonia?" *IEEE Access*, Vol. 8, 2020, pp. 132665 - 132676.
- [13] Rahman, T., Khandakar, A., Qiblawey, Y., Tahir, A., Kiranyaz, S., Kashem, S.B.A., Islam, M.T., Maadeed, S.A., Zughair, S.M., Khan, M.S. and Chowdhury, M.E., 2020. Exploring the Effect of Image Enhancement Techniques on COVID-19 Detection using Chest X-ray Images.