Universidade Estadual de Londrina

CLAUDIO BENTO REIS NETO

USO DE REDES NEURAIS CONVOLUCIONAIS NO DIAGNÓSTICO DE COVID-19 EM IMAGENS DE RAIO-X DO TÓRAX

LONDRINA
2023

CLAUDIO BENTO REIS NETO

USO DE REDES NEURAIS CONVOLUCIONAIS NO DIAGNÓSTICO DE COVID-19 EM IMAGENS DE RAIO-X DO TÓRAX

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Ms. Vanessa Matias Leite

LONDRINA
2023

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UEL

Sobrenome, Nome.
Título do Trabalho : Subtitulo do Trabalho / Nome Sobrenome. - Londrina, 2017.
100 f. : il.

Orientador: Nome do Orientador Sobrenome do Orientador.
Coorientador: Nome Coorientador Sobrenome Coorientador.
Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Centro de Ciências Exatas, Programa de Pós-Graduação em Ciência da Computação, 2017.
Inclui bibliografia.

1. Assunto 1 - Tese. 2. Assunto 2 - Tese. 3. Assunto 3 - Tese. 4. Assunto 4 - Tese. I. Sobrenome do Orientador, Nome do Orientador. II. Sobrenome Coorientador, Nome Coorientador. III. Universidade Estadual de Londrina. Centro de Ciências Exatas. Programa de Pós-Graduação em Ciência da Computação. IV. Título.

CLAUDIO BENTO REIS NETO

USO DE REDES NEURAIS CONVOLUCIONAIS NO DIAGNÓSTICO DE COVID-19 EM IMAGENS DE RAIO-X DO TÓRAX

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA

Orientador: Prof. Ms. Vanessa Matias Leite
Universidade Estadual de Londrina

Coorientador: Prof. Dr. Guilherme Pina Cardim
Universidade Estadual Paulista – UNESP

Prof. Dr. Terceiro Membro da Banca
Universidade/Instituição do Terceiro Membro da Banca – Sigla instituição

Londrina, 03 de maio de 2023.

Dedico esse trabalho à todos que me incentivaram a entrar na universidade e a não desistir dessa conquista.

AGRADECIMENTOS

Agradeço ao meu orientador Guilherme por ter me introduzido ao mundo da Computação Gráfica e por sempre me incentivar e esclarecer minhas dúvidas durante o trabalho. Também agradeço à minha orientadora Vanessa por sempre estar disposta a me ajudar e sanar minhas dúvidas.

Agradeço à Universidade Estadual de Londrina e aos professores do departamento de computação por terem me fornecido toda a base necessária para chegar até aqui.

Agradeço a todos os amigos que fiz durante estes anos, em especial a Felipe Barusso, Gabriel Sabaudo, Guilherme Silva e Vinícius Cesar, pela amizade que foi fundamental para me manter motivado e perseverante durante este período.

Agradeço à minha namorada Renata por sempre estar ao meu lado, especialmente nos momentos difíceis, quando me deu forças para continuar.

Por fim, agradeço à minha família, que sempre me apoiou e fez tudo o que estava ao seu alcance para que eu pudesse ingressar na universidade.

"If you believe in nothing else.
Just keep believing in yourself.
There will be times of trouble, it's gonna
hurt like hell.
This much I know, all ends well.
It all ends well"
All Ends Well (Alter Bridge, 2013)

NETO, C. B. R.. **Uso de Redes Neurais Convolucionais no diagnóstico de COVID-19 em imagens de Raio-x do Tórax**. 2023. 37f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Universidade Estadual de Londrina, Londrina, 2023.

RESUMO

A pandemia da COVID-19 é um evento que será lembrado para sempre. Desde o seu início, tem-se buscado um diagnóstico rápido e econômico. Nesse sentido, este trabalho consiste na aplicação e análise comparativa do desempenho de diferentes redes neurais na detecção de possíveis diagnósticos positivos de COVID-19 em imagens de exames de raio-X do tórax, utilizando Processamento digital de imagens e quatro redes neurais: VGG16, VGG19, ResNet50 e ResNet50V2. Foi observado que, embora a ResNet50V2 tenha obtido a melhor acurácia, com 95,55%, ao analisar os tipos de erro, a ResNet50 obteve o resultado mais confiável para realizar a classificação.

Palavras-chave: COVID-19. Classificação. Imagem. Inteligência artificial. Rede Neural. Análise. Comparação

NETO, C. B. R.. **Use of Convolutional Neural Networks in the diagnosis of COVID-19 in Chest X-ray images**. 2023. 37p. Final Project (Bachelor of Science in Computer Science) – State University of Londrina, Londrina, 2023.

ABSTRACT

The COVID-19 pandemic is an event that will be remembered forever. Since its beginning, there has been a search for a quick and economical diagnosis. In this sense, this work consists of the application and comparative analysis of the performance of different neural networks in the detection of possible positive COVID-19 diagnoses in chest X-ray images, using Digital Image Processing and four neural networks: VGG16, VGG19, ResNet50, and ResNet50V2. Although ResNet50V2 achieved the best accuracy, at 95.55%, when analyzing the types of errors, ResNet50 obtained the safest result for performing the classification.

Keywords: COVID-19. Classification. Image. Artificial Inteligence. Neural Network. Analysis. Comparison

LISTA DE ILUSTRAÇÕES

Figura 1 – Imagem do raio-X do tórax (Fonte: [1]). 15
Figura 2 – Sequência de etapas do Processamento digital de imagens (Fonte adaptada de [2]). 16
Figura 3 – Representação gráfica de um exemplo de rede neural artificial. 17
Figura 4 – Representação gráfica da arquitetura de uma rede neural convolucional (Fonte adaptada de [3]). 18
Figura 5 – Representação gráfica da camada de convolução. 18
Figura 6 – Representação gráfica de um *max pooling* e de um *average pooling*. . . . 19
Figura 7 – Representação gráfica do modelo VGG16 (Fonte adaptada de [4]). . . . 20
Figura 8 – Representação gráfica do modelo VGG19 (Fonte adaptada de [4]). . . . 21
Figura 9 – Representação gráfica da estrutura de um bloco residual (Fonte adaptada de [5]). 22
Figura 10 – Representação gráfica do modelo ResNet50 (Fonte adaptada de [6]). . . 22
Figura 11 – Exemplos de imagens usadas na classificação (Fonte: [1]). 26
Figura 12 – Acurácia do treino e validação dos modelos treinados com 150 épocas. . 30
Figura 13 – Matrizes de confusão dos modelos. 30
Figura 14 – Exemplos de predições do modelo ResNet50V2. 32
Figura 15 – Exemplo de predições incorretas dos modelos VGG16 e VGG19. 33
Figura 16 – Predição incorreta do modelo ResNet50. 33

LISTA DE TABELAS

Tabela 1 – Matriz de confusão. 24
Tabela 2 – Tamanho dos pesos utilizados em cada modelo (Fonte: [7]). 28
Tabela 3 – Valores de acurácia média de treino e validação. 29
Tabela 4 – Valor de acurácia dos modelos treinados. 31
Tabela 5 – Valores de precisão, revocação e pontuação F1 dos modelos treinados. . 31

LISTA DE ABREVIATURAS E SIGLAS

ADAM	*Adaptive Moment Estimation*
AM	Aprendizado de máquina
OMS	Organização Mundial da Saúde
PDI	Processamento digital de imagens
ReLu	*Rectified Linear Unit*
RNC	Rede neural convolucional
RNA	Rede neural artificial
ResNet	*Residual Neural Network*
VGG	*Visual Geometry Group*

SUMÁRIO

1 INTRODUÇÃO . . . 13

2 FUNDAMENTAÇÃO TEÓRICA . . . 14
2.1 COVID-19 . . . 14
2.1.1 O uso de imagem no diagnóstico de COVID-19 . . . 14
2.2 Processamento digital de imagens . . . 15
2.3 Aprendizado de máquina . . . 16
2.3.1 Rede neural artificial . . . 16
2.3.2 Rede neural convolucional . . . 17
2.3.2.1 Arquitetura de uma RNC . . . 17
2.3.2.2 Funções de ativação . . . 19
2.4 VGG . . . 20
2.4.1 VGG16 . . . 20
2.4.2 VGG19 . . . 20
2.5 ResNet . . . 21
2.5.1 ResNet50 . . . 22
2.5.2 ResNet50V2 . . . 23
2.6 Cálculo de desempenho e métricas . . . 23
2.6.1 Matriz de confusão . . . 24
2.6.2 Acurácia . . . 24
2.6.3 Precisão . . . 24
2.6.4 Revocação . . . 24
2.6.5 Pontuação F1 . . . 25

3 DESENVOLVIMENTO . . . 26
3.1 Seleção de imagens . . . 26
3.2 Pré-processamento . . . 27
3.3 Implementação das redes neurais . . . 27

4 RESULTADOS . . . 29
4.1 Resultado dos treinos . . . 29
4.2 Resultado dos testes . . . 30
4.3 Análise dos resultados . . . 31

5 CONCLUSÃO . . . 35

REFERÊNCIAS . . . 36

1 INTRODUÇÃO

Desde o início de 2020, a sociedade vem tendo que conviver com a pandemia da COVID-19 [8]. A doença causada pelo vírus SARS-CoV-2 infectou, e ainda infecta, milhões de pessoas, inclusive, podendo levar à morte.

O diagnóstico da doença respiratória sempre foi um desafio. Neste caso, um diagnóstico rápido e barato é primordial, levando em consideração o grande número de casos e o alto contágio. O primeiro método usado para diagnosticar a doença foi por meio de um exame de raio-X do Tórax. O motivo é o fato do raio-X ser um exame acessível, pois já era muito usado para diagnóstico de outras doenças [9].

Um raio-x do tórax de uma pessoa doente pode incluir áreas brancas difusas que indicam acúmulo de líquido e inflamação nos pulmões. Em casos mais graves, os achados podem incluir áreas de colapso pulmonar, que se manifesta pela diminuição da expansão do pulmão e pode aparecer como uma área escura em um raio-x do tórax [10].

A partir disso, surgiram vários trabalhos com o objetivo de detectar indícios de COVID-19 em imagens de raio-x do tórax utilizando inteligência artificial. Uma das técnicas amplamente utilizadas nesse objetivo são as redes neurais, que consistem em um modelo computacional composto por camadas de neurônios interconectados, capazes de processar informações e aprender a partir de dados de entrada para produzir saídas precisas. As redes neurais são amplamente utilizadas em tarefas de reconhecimento de padrões, classificação e predição em imagens [11].

Considerando essa demanda, este trabalho propôs a aplicação e análise comparativa do desempenho de diferentes redes neurais na detecção de possíveis diagnósticos positivos de COVID-19 em exames de raio-X do tórax. Para isso, será usado um conjunto de imagens de exames, Processamento digital de imagens [2] e o uso das redes pré-treinadas: VGG16, VGG19, ResNet50 e ResNet50V2.

O objetivo deste trabalho consiste em comparar os resultados obtidos pelas redes neurais e verificar qual delas é a mais indicada para realizar a classificação de imagens relacionadas à detecção da COVID-19, contribuindo assim, para as áreas de pesquisa relacionadas ao diagnóstico da doença por meio de imagens.

O restante do trabalho está organizado da seguinte forma: o Capítulo 2 apresenta os principais conceitos utilizados no desenvolvimento do trabalho. O Capítulo 3 descreve em detalhes o desenvolvimento, incluindo a seleção e preparação das imagens, e a implementação das redes neurais. O Capítulo 4 apresenta os resultados obtidos, incluindo a análise do desempenho de cada rede neural testada. Por fim, no Capítulo 5, são apresentadas as conclusões do trabalho, bem como sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

A fundamentação teórica apresenta conceitos importantes presentes no trabalho, começando por uma breve apresentação sobre a COVID-19 e o uso de exames de imagens para detectá-lo, seguida da definição de Processamento digital de imagens e suas etapas. Em seguida, são apresentados os conceitos de Aprendizado de máquina, redes neurais e seus elementos, além da apresentação das redes neurais utilizadas no trabalho: VGG16, VGG19, ResNet50 e ResNet50V2. Por fim, são apresentadas as métricas de avaliação utilizadas para avaliar o desempenho dos modelos de redes neurais.

2.1 COVID-19

Em dezembro de 2019, um surto de pneumonia de origem desconhecida foi relatado em Wuhan, cidade de Hubei, província da China [12]. Sua origem foi descoberta pouco tempo depois, quando descobriu-se que a doença é causada pelo vírus SARS-CoV-2. Após isso, denominou-se a doença como coronavirus (COVID-19).

No dia 12 de março de 2020, a disseminação mundial do SARS-CoV-2 e as milhares de mortes causadas levaram a Organização Mundial da Saúde (OMS) a declarar uma pandemia. [8].

2.1.1 O uso de imagem no diagnóstico de COVID-19

Exames de imagem foram inicialmente utilizados como principal ferramenta para diagnóstico de COVID-19. Isso se deve ao fato do alto custo de outras formas de diagnóstico e à alta taxa de ocorrência da doença ao redor do planeta [9].

Atualmente, um dos principais protocolos para casos de suspeita da doença com urgência de um diagnóstico, é a utilização de imagens de Tomografia Computadorizada ou de raio-X do tórax. A principal vantagem da utilização deste exame é o seu fácil acesso, pelo fato dele já ser utilizado no diagnóstico de várias outras doenças [9].

Outro fator pelo qual exames de imagens são utilizados no diagnóstico, é o fato de que aproximadamente 59% dos pacientes doentes apresentam alterações no exame de imagem. No entanto, um exame que não acusa nenhuma alteração não pode ser usado como única fonte para se afirmar que a pessoa não está infectada, pois somente o exame de imagem sem alteração não é fator de exclusão [9].

Para efeito de exemplificação, a Figura 1 mostra um exame de raio-X do tórax, que pode ser usado para diagnosticar COVID-19.

Figura 1 – Imagem do raio-X do tórax (Fonte: [1]).

2.2 Processamento digital de imagens

Processamento digital de imagens (PDI) é um conjunto de técnicas de processamento de dados multidimensionais que recebem uma imagem como entrada. O PDI, na maioria dos casos, não é uma tarefa trivial, pois envolve um conjunto de tarefas interconectadas que vão desde a captura da imagem, até a interpretação e reconhecimento dos objetos [2].

O PDI pode estar atrelado a alguma técnica de Aprendizado de máquina e ao reconhecimento de padrões, tornando o processo automatizado.

Podemos citar como etapas do PDI:

1. Aquisição de imagens: obter a imagem de entrada para o processamento.

2. Pré-processamento: composta por técnicas que possuem como objetivo melhorar a qualidade da imagem e envolvem duas categorias: métodos que operam no domínio espacial (baseadas em filtros que manipulam o plano da imagem) e métodos que operam no domínio da frequência (baseadas em filtros que agem sobre o espectro da imagem) [2].

3. Segmentação: utilizada para separar a imagem nas partes que a constituem e que se diferenciam entre si. É usual denominar os grupos de *pixels* de interesse como objetos.

4. Pós-processamento: etapa na qual os principais defeitos e imperfeições da segmentação são corrigidos.

5. Extração de características: nesta etapa é feita a extração das informações consideradas úteis da imagem.

6. Reconhecimento e interpretação: possui como objetivo identificar os objetos segmentados na imagem e interpretá-los para se chegar a um resultado.

A Figura 2 apresenta uma representação gráfica da sequência de etapas do PDI.

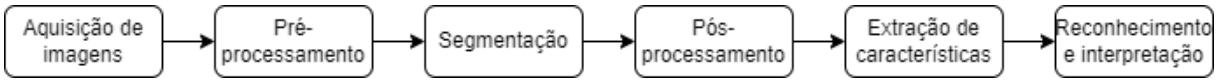


Figura 2 – Sequência de etapas do Processamento digital de imagens (Fonte adaptada de [2]).

2.3 Aprendizado de máquina

Aprendizado de máquina (AM) é a ciência que tem como objetivo fazer a análise de dados e criar modelos computacionais de aprendizado. É considerado um ramo de algoritmos computacionais projetados para emular a inteligência humana aprendendo com o ambiente ao redor [13].

Os algoritmos de AM podem ser classificados como "*soft-coded*", ou seja, não precisam ser literalmente programados para alcançar a tarefa desejada e chegar a um resultado. Esta classificação pode ser explicada pelo fato desse tipo de algoritmo se adaptar ou alterar sua arquitetura por meio da repetição, para que se torne cada vez melhor na realização da tarefa. Este processo é conhecido como treinamento, no qual são fornecidos dados de entrada juntamente com os resultados desejados. O algoritmo, então, se configura para tentar atingir os resultados esperados [13].

2.3.1 Rede neural artificial

Uma rede neural artificial (RNA) é um método da inteligência artificial que processa dados de forma inspirada no cérebro humano. Dentro do AM, este processo é chamado de aprendizado profundo e usa nós ou neurônios interconectados em uma estrutura de camadas, semelhante ao cérebro humano. Um neurônio é um objeto de uma rede neural que recebe dados e os pesos de cada ligação [14]. O gerenciamento destes pesos é realizado por um processo denominado treinamento, o qual é responsável pela extração e classificação das características dos dados de entrada da rede [15].

Para implementar uma RNA, é necessário definir algumas variáveis, tais como:

1. O número de nós (neurônio) da camada de entrada.

2. O número de camadas ocultas e o número de neurônios a serem colocados nestas camadas.

3. O número de neurônios na camada de saída.

Na Figura 3, está representado graficamente um exemplo de uma rede neural artificial.

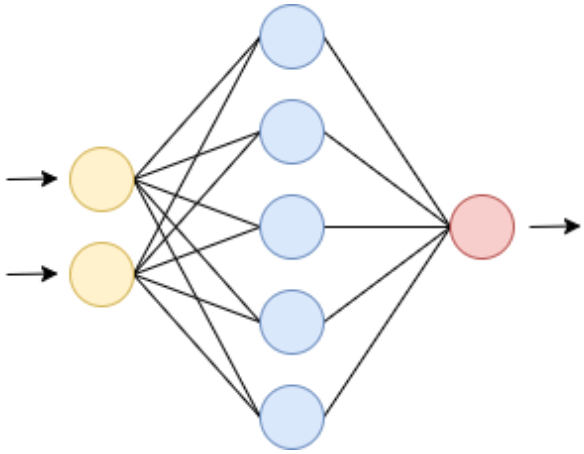


Figura 3 – Representação gráfica de um exemplo de rede neural artificial.

Normalmente, as RNAs são utilizadas na resolução de problemas em que o comportamento das variáveis não é completamente conhecido. Uma de suas principais características é a capacidade de aprender por meio de exemplos e generalizar a informação aprendida [11].

2.3.2 Rede neural convolucional

Uma rede neural convolucional (RNC) é um tipo de RNA que foi proposta para facilitar o trabalho com imagens. Atualmente, é o modelo mais eficiente no que se refere à classificação de dados de imagens [16]. Isso se deve ao fato das suas camadas reduzirem a alta dimensionalidade das imagens sem perder suas informações, por meio da otimização da saída [3].

Uma RNC é similar a uma RNA clássica no que se diz respeito a manipulação de dados, porém, apresenta alguns benefícios. O aspecto de maior benefício das RNCs é a redução do número de parâmetros a serem treinados [7]. Esse fato fez com que tarefas mais complexas pudessem ser abordadas com modelos de RNCs, o que não era possível com as RNAs clássicas.

2.3.2.1 Arquitetura de uma RNC

Como as RNAs, as RNCs também são compostas por uma camada de entrada, uma camada de saída e várias camadas ocultas entre elas. Na Figura 4, é possível observar graficamente a arquitetura de uma RNC, com suas diferentes camadas.

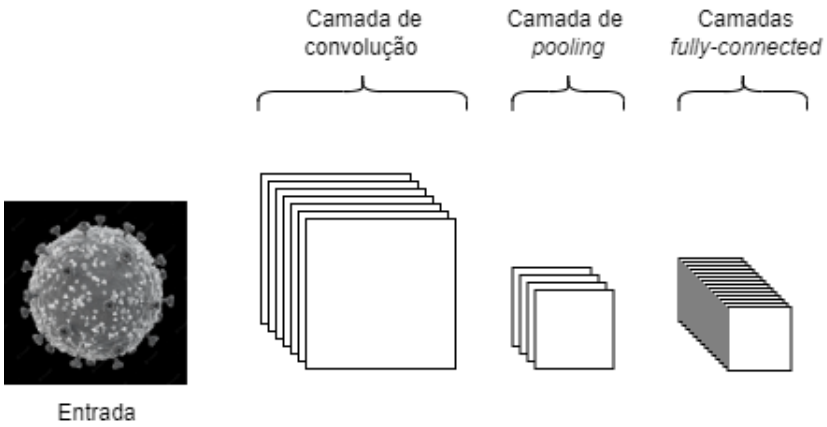


Figura 4 – Representação gráfica da arquitetura de uma rede neural convolucional (Fonte adaptada de [3]).

A camada de convolução é parte fundamental no funcionamento de uma RNC. Nela, são definidos núcleos de tamanho geralmente menor do que a imagem de entrada, preenchidos com valores. A saída de uma camada de convolução é calculada, primeiramente, por meio da sobreposição do núcleo na entrada e depois com a soma da multiplicação dos valores do núcleo com o *pixel* equivalente da imagem. Na Figura 5 é possível observar mais claramente o funcionamento da camada sendo aplicada em uma imagem de tamanho 3x3 com núcleo menor, de tamanho 2x2.

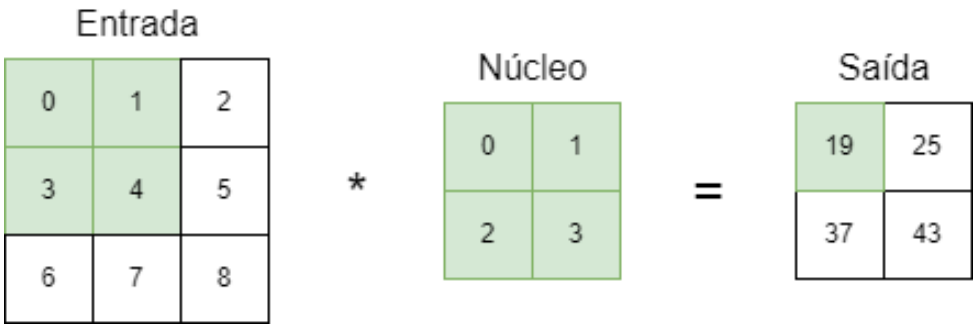


Figura 5 – Representação gráfica da camada de convolução.

A camada de *pooling* tem como função diminuir o tamanho da imagem, reduzindo assim, o processamento computacional necessário para se trabalhar com ela [3]. Existem alguns tipos de *pooling*, sendo que o *max polling* e o *average polling* são os mais populares. No *max pooling*, o algoritmo buscará o valor máximo dentre os encontrados no filtro e no *average polling* a média dos valores encontrados será aplicada. No caso apresentado na Figura 6, os dois filtros de tamanho 2x2 são aplicados em uma imagem 4x4. O tamanho do filtro determina a região da imagem onde ele será aplicado e o tamanho da imagem de saída.

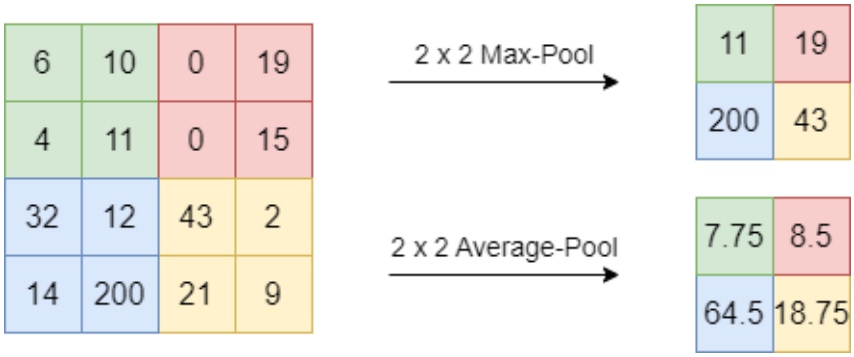


Figura 6 – Representação gráfica de um *max pooling* e de um *average pooling*.

Por fim, as camadas *fully-connected* executam as mesmas tarefas encontradas nas RNAs clássicas, a classificação dos dados de entrada.

2.3.2.2 Funções de ativação

Em uma RNC, as funções de ativação são usadas para transformar um sinal de entrada em um sinal de saída, que por sua vez é alimentado como entrada para a próxima camada da rede. A ativação é aplicada à soma ponderada da entrada e dos pesos correspondentes. A saída da camada é obtida após a aplicação da função de ativação, e essa saída é então fornecida como entrada para a próxima camada [11].

As funções de ativação são usadas para introduzir a não-linearidade no modelo de RNC, para que a rede possa aprender progressivamente representações de recursos mais eficazes [17]. Pode-se citar como exemplo: a Linear, *Sigmoid*, *Rectified Linear Unit* (ReLu) e *Softmax*. O foco deste trabalho são as funções ReLU e *Softmax*, as quais foram utilizadas em seu desenvolvimento.

ReLu é uma função de ativação amplamente utilizada. Sua principal vantagem é que ela ajuda a prevenir o problema de desvanecimento do gradiente, comum em redes neurais profundas, pois mantém o gradiente constante em regiões positivas [18]. Além disso, a ReLu desativa um neurônio quando a saída da transformação linear é menor ou igual a zero, o que ajuda a reduzir a complexidade computacional [19]. Isso ocorre porque a saída da função é zero para todos os valores de entrada negativos e a saída é igual à entrada para valores positivos. A equação 2.1 representa matematicamente a ReLu [19].

$$f(x) = max(0, x) \tag{2.1}$$

Softmax é uma função de ativação comumente utilizada para classificar os resultados de saída de uma rede neural. Ela retorna valores no intervalo de 0 a 1 que podem ser interpretados como probabilidades dos dados analisados pertencerem a uma determinada classe. A equação 2.2 representa matematicamente a *Softmax* [19].

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \tag{2.2}$$

2.4 VGG

Visual Geometry Group (VGG) é um grupo de pesquisa em visão computacional e AM sediado na Universidade de Oxford, no Reino Unido [4]. O VGG é conhecido por diversos trabalhos inovadores em RNCs, incluindo a VGGNet (ou VGG16), umas das primeiras RNCs profundas com muitas camadas. Após ela, surgiram várias outras, incluindo a VGG19.

2.4.1 VGG16

VGG16 é um modelo de RNC desenvolvido pelos pesquisadores A. Zisserman e K. Simonyan, da Universidade de Oxford [4]. O modelo suporta 16 camadas, sendo 13 camadas convolucionais e 3 camadas *fully connected*. A VGG16 é capaz de classificar imagens em 1000 categorias diferentes de objetos, animais, veículos, alimentos, entre outros. A Figura 7 representa graficamente a arquitetura da VGG16.

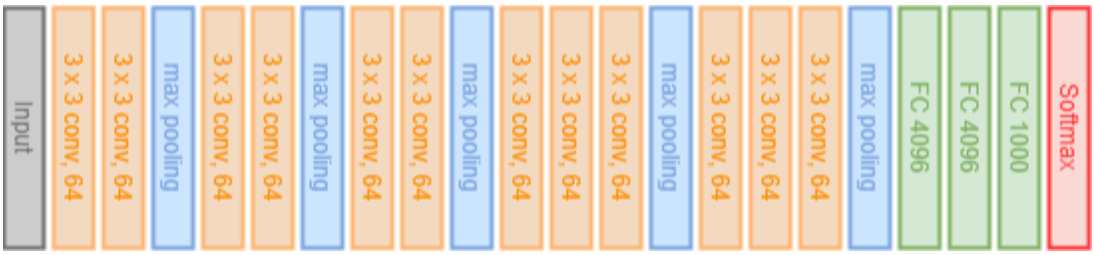


Figura 7 – Representação gráfica do modelo VGG16 (Fonte adaptada de [4]).

A RNC recebe como entrada uma imagem de tamanho 224x224 *pixels* e, em seguida, aplica uma série de camadas de convolução com um núcleo de tamanho 3x3, 64 filtros e ativação ReLu, intercaladas por camadas de *max pooling* para reduzir o tamanho da imagem. Após isso, são adicionadas algumas camadas *fully connected* com ativação ReLu para classificar os dados, seguidas por uma camada com a função de ativação *Softmax* para calcular a probabilidade de cada classe e classificar a imagem de entrada [20].

2.4.2 VGG19

A VGG19 é um modelo de RNC com um conceito muito similar ao da VGG16. A principal diferença entre eles é que a VGG19 possui suporte a 19 camadas, sendo 16 camadas convolucionais e 3 camadas *fully connected*, sendo que as especificações das camadas não diferem.

A VGG19 foi proposta em 2014 e é amplamente utilizada como referência em tarefas de classificação de imagens. A seguir, na Figura 8, está representada graficamente a arquitetura da VGG19.

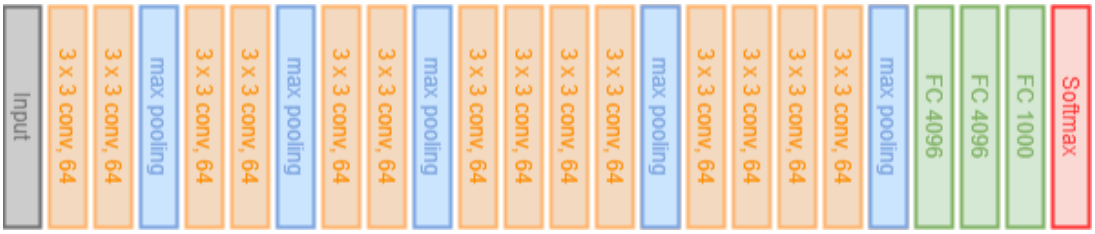


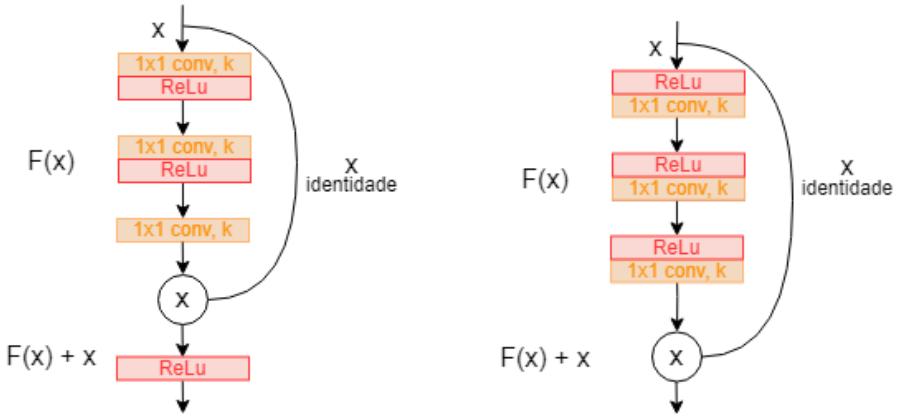
Figura 8 – Representação gráfica do modelo VGG19 (Fonte adaptada de [4]).

2.5 ResNet

Residual Neural Network (ResNet) é uma arquitetura de RNC desenvolvida pela *Microsoft Research* em 2015 [6] e possui duas versões, ResNet e ResNetV2. Foi projetada para solucionar o problema de desvanecimento do gradiente, que ocorre quando a rede fica muito profunda e as atualizações de peso durante o treinamento se tornam cada vez menores. A solução para este problema foi proposta utilizando blocos residuais, que permitem que as informações de entrada fluam diretamente para as camadas de saída, contornando as camadas intermediárias [6].

A estrutura do bloco residual difere de acordo com a versão do modelo. Na primeira versão, representada na Figura 9 (a), o caminho principal do bloco residual é composto por três camadas de convolução, normalmente com um número reduzido de filtros, seguidas por uma camada de ativação ReLu. Já na Figura 9 (b), está representada a estrutura do bloco residual da versão ResNetV2, na qual a ativação ReLu antecede as camadas de convolução.

Em ambas as versões, além do caminho principal, o bloco residual apresenta um atalho, permitindo que a informação flua diretamente da entrada para a saída do bloco. Na prática, o atalho permite que a rede neural salte uma ou mais camadas convolucionais do caminho principal, o que pode acelerar o processo de treinamento e melhorar a precisão do modelo [6].



(a) Bloco residual da ResNet. (b) Bloco residual da ResNetV2.

Figura 9 – Representação gráfica da estrutura de um bloco residual (Fonte adaptada de [5]).

2.5.1 ResNet50

ResNet50 é um modelo de RNC projetado para permitir o treinamento de redes mais profundas com maior facilidade, evitando o problema de desempenho degradante que ocorre com o aumento da profundidade da rede. Para isso, utiliza os blocos residuais já citados anteriormente.

A ResNet50 suporta 50 camadas e foi treinada em um grande conjunto de dados de imagens. Normalmente, é utilizada como base para redes mais complexas, com muitas categorias. A Figura 10 representa graficamente a arquitetura da ResNet50.

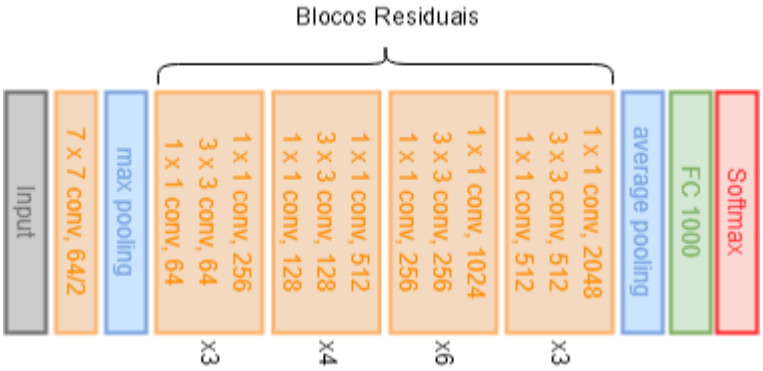


Figura 10 – Representação gráfica do modelo ResNet50 (Fonte adaptada de [6]).

A arquitetura da ResNet50 também se inicia com a entrada (*input*) de uma imagem de tamanho 224x224 *pixels* e após isso é divida em 5 estágios. São eles: [6]

- Estágio 1: consiste em uma única camada convolucional com 64 filtros e núcleo de tamanho 7x7. Essa camada é utilizada para reduzir o tamanho da imagem de entrada.

- Estágio 2: consiste em 3 blocos residuais, cada um contendo 3 camadas convolucionais: duas com 64 filtros e uma com 256 filtros. Essas camadas são utilizadas para reduzir ainda mais o tamanho da imagem e extrair características de baixo nível.
- Estágio 3: consiste em 4 blocos residuais, cada um contendo 3 camadas convolucionais: duas com 128 filtros e uma com 512 filtros. Essas camadas são utilizadas para extrair características de nível médio.
- Estágio 4: consiste em 6 blocos residuais, cada um contendo 3 camadas convolucionais: duas com 256 filtros e uma com 1024 filtros. Essas camadas são utilizadas para extrair características de nível alto.
- Estágio 5: consiste em 3 blocos residuais, cada um contendo 3 camadas convolucionais: duas com 512 filtros e uma com 2048 filtros. Essas camadas são utilizadas para refinar ainda mais as características extraídas.

Entre os estágios 1 e 2, a RNC passa por uma camada de *max pooling*. Após o estágio 5, a rede passa por uma camada de *average pooling*, seguida de uma camada *fully connected* e, por fim, pelo *Softmax* para classificar os dados de saída.

2.5.2 ResNet50V2

O ResNet50V2 é um modelo de RNC considerado a evolução do ResNet50. Mesmo tendo conceitos similares, a versão mais atualizada do modelo tem melhor eficiência e precisão.

Apesar de possuírem arquiteturas similares, a principal diferença da ResNet50V2 para sua antecessora é a utilização da versão atualizada do bloco residual, que inclui uma camada de ativação antes da camada convolucional em cada bloco. Além disso, o ResNet50V2 utiliza um esquema de inicialização de peso diferente e uma técnica de atenuação de ativação para melhorar ainda mais a eficiência do modelo [6].

2.6 Cálculo de desempenho e métricas

Para se avaliar o desempenho de uma rede neural é necessário definir métricas a serem utilizadas. Essas métricas são muito importantes porque permitem que seja avaliado o quão bem a rede neural está funcionando em relação aos dados de treinamento e de teste.

Existem muitas métricas de avaliação diferentes, suas utilizações dependem do tipo do problema que está sendo resolvido. Na literatura, as métricas mais comumente utilizadas são as matrizes de confusão, acurácia, precisão, revocação e a pontuação F1.

2.6.1 Matriz de confusão

A matriz de confusão é uma tabela que mostra a frequência com que as previsões feitas pelo modelo correspondem ou não às classes reais dos exemplos. Com os valores obtidos através da matriz de confusão, é possível calcular as outras métricas.

A Tabela 1 mostra onde as saídas, reais e preditas, devem ser colocadas para serem classificadas. As possíveis classificações são: verdadeiro positivo (VP), verdadeiro negativo (VN), falso positivo (FP) e falso negativo (FN). Onde, verdadeiro ou falso indica se a predição da rede neural está correta. Positivo ou negativo indica qual foi o valor predito pela rede.

		Valores Preditos	
		Sim	**Não**
Valores Reais	**Sim**	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Tabela 1 – Matriz de confusão.

2.6.2 Acurácia

Acurácia é a proporção de predições corretas que a rede neural faz, sendo elas positivas ou negativas. O cálculo da acurácia é realizado conforme apresentado na Equação 2.3.

$$acuracia = \frac{VP + VN}{VP + FP + FN + VN} \tag{2.3}$$

2.6.3 Precisão

Precisão é a medida que indica o quão corretamente a rede neural é capaz de classificar ou prever os dados de entrada. O cálculo da precisão é realizado conforme apresentado na Equação 2.4.

$$precisao = \frac{VP}{VP + FP} \tag{2.4}$$

2.6.4 Revocação

Revocação é uma medida usada para avaliar a capacidade da rede neural em detectar todas as amostras de uma classe específica. O cálculo da revocação é realizado conforme apresentado na Equação 2.5.

$$revocacao = \frac{VP}{VP + FN} \tag{2.5}$$

2.6.5 Pontuação F1

Pontuação F1 é uma medida importante para se avaliar a qualidade de um modelo de classificação, especialmente quando as classes são desequilibradas. Uma pontuação F1 alta indica que o modelo tem um bom desempenho na classificação das amostras em suas respectivas classes. O cálculo da pontuação F1 é realizado conforme apresentado na Equação 2.6.

$$F1 = 2 \times \frac{precisao \times revocacao}{precisao + revocacao} = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{2.6}$$

3 DESENVOLVIMENTO

O desenvolvimento do trabalho foi dividido em três etapas. A primeira consistiu na seleção das imagens que foram utilizadas como entrada para as redes neurais. Em seguida, foi realizado um pré-processamento nas imagens selecionadas. Por fim, foi feita a implementação das redes neurais utilizando as imagens pré-processadas.

3.1 Seleção de imagens

A seleção de imagens foi realizada utilizando o banco de dados "COVID-19 *Radiography Database*" [1], o qual conta com mais de 15000 imagens de raio-X de tórax com as mais variadas características, incluindo se há indícios de COVID-19, Pneumonia ou se não há indícios de quaisquer doenças.

As imagens disponíveis no banco foram capturadas de maneira diferente uma das outras, pois possuem diversas fontes de obtenção de exames. Um exemplo disso é a existência de imagens de exames de raio-X frontais e laterais. Portanto, somente as imagens de exames frontais foram selecionadas para a classificação.

Após a filtragem, foram selecionadas 1000 imagens, sendo que 500 são de exames que obtiveram indícios de COVID-19 e 500 de exames sem nenhum indício da doença. A Figura 11 (a) mostra uma imagem de um exame frontal com indícios de COVID-19, já a Figura 11 (b) mostra uma imagem de um exame que também é frontal, mas não contém indícios da doença.

(a) Exame frontal com indícios de COVID-19.

(b) Exame frontal sem indícios de COVID-19.

Figura 11 – Exemplos de imagens usadas na classificação (Fonte: [1]).

3.2 Pré-processamento

Para que as imagens possam ser utilizadas como entrada para as redes neurais, foi necessário realizar um pré-processamento utilizando a biblioteca *OpenCV*, em *Python*. Nesse processo, todas as imagens foram redimensionadas para 224x224 *pixels*.

Após esse processo, as imagens são divididas de acordo com sua classe, neste trabalho chamaremos de "Covid" a classe contendo as imagens de exames com indícios de COVID-19 e classe "Normal" a que contém imagens de exames sem indícios da doença.

Por fim, as imagens são divididas em três conjuntos: 70% pertencem ao conjunto de treino, 10% ao conjunto de validação e 20% ao conjunto de teste. O conjunto de treino é utilizado para o treinamento da rede neural. Já o conjunto de validação é utilizado para avaliar a eficácia da rede durante o processo de treinamento, utilizando um conjunto diferente do conjunto de treino. Por fim, o conjunto de teste é utilizado após o treinamento da rede para calcular a acurácia obtida pelo modelo treinado.

3.3 Implementação das redes neurais

Os modelos de RNC escolhidos para serem implementados foram: VGG16, VGG19, ResNet50 e ResNet50V2. Todas elas são redes pré-treinadas, implementadas com a linguagem de programação *Python* e a biblioteca Keras [7], amplamente utilizada na construção de redes neurais profundas. As redes foram executadas utilizando a biblioteca de código aberto *Tensorflow*, por meio de uma máquina virtual fornecida pelo *Google Colaboratory* (Colab) [21].

Todas as redes neurais foram pré-treinadas com os pesos disponibilizados pelo *ImageNet* [22], um grande conjunto de dados de imagens desenvolvido para ser utilizado em tarefas de visão computacional, tais como reconhecimento e detecção de objetos e classificação de imagens. O uso desses pesos pode melhorar significativamente o desempenho da RNC em uma tarefa específica, reduzindo o tempo de treinamento necessário para alcançar bons resultados.

O tamanho dos pesos afeta diretamente o número de parâmetros do modelo, o que pode influenciar a quantidade de memória e energia necessárias para executá-lo durante o treinamento. Modelos com mais parâmetros (pesos) geralmente possuem maior capacidade para aprender e memorizar padrões complexos, mas podem ser mais difíceis de treinar e exigir mais recursos computacionais, como memória e poder de processamento. Por outro lado, modelos com menos parâmetros podem ser mais fáceis de treinar, mas podem não ter a mesma capacidade para capturar a complexidade dos dados de entrada. A Tabela 2 apresenta o tamanho dos pesos usados em cada modelo de RNC.

Modelo	**Tamanho dos pesos**
VGG16	528 MB
VGG19	549 MB
ResNet50	98 MB
ResNet50V2	98 MB

Tabela 2 – Tamanho dos pesos utilizados em cada modelo (Fonte: [7]).

Os modelos foram configurados com hiperparâmetros específicos para atender às necessidades do problema em questão, que possui duas classes distintas. Foi escolhida a função de ativação *Softmax* para a camada de saída, uma vez que é amplamente utilizada quando se deseja atribuir uma probabilidade para cada classe. Quanto à função de custo, optou-se pela *Categorical Crossentropy*, uma das principais funções utilizadas na literatura para obtenção do valor de custo em problemas com múltiplas classes.

Alguns dos hiperparâmetros foram definidos por meio de testes durante o processo de treinamento dos modelos, a fim de alcançar um bom desempenho em todas as redes. Foi utilizada a técnica de otimização *Adaptive Moment Estimation* (ADAM) e um valor de 0,6 para o *dropout*, a fim de evitar o sobreajuste (*overfitting*), que ocorre quando um modelo se adapta demais aos dados de treinamento e não consegue generalizar padrões e características úteis para dados de teste [23].

A taxa de aprendizado (*learning rate*) determina a rapidez com que o modelo ajusta os pesos durante o treinamento, e foi definida como 1×10^{-3}. Já o tamanho do lote (*batch size*) define o número de amostras de treinamento usadas em cada atualização de peso e foi definido como 32. A escolha de ambos os valores foi realizada buscando um equilíbrio entre desempenho e poder computacional necessário.

Para determinar o número de épocas adequado, foram realizados testes com os valores de 75, 100 e 150. O objetivo foi escolher o valor que apresentasse o melhor desempenho para os modelos de RNC testados. Testes com um número maior de épocas também foram realizados, porém verificou-se que não houve uma melhora de desempenho significativa que justificasse o aumento no poder computacional exigido.

Após a finalização da definição dos hiperparâmetros, inicia-se o processo de treinamento das redes neurais utilizando os conjuntos de treino e validação. Posteriormente, para avaliar o desempenho do modelos treinados, o conjunto de teste é utilizado a fim de gerar os valores das métricas de avaliação.

4 RESULTADOS

Os resultados do trabalho serão apresentados em três etapas. Primeiramente, serão apresentados os resultados do processo de treinamento de cada modelo. Em seguida, serão apresentados os resultados do processo de teste de cada modelo. Por fim, será realizada a análise dos resultados e comparação entre os modelos selecionados.

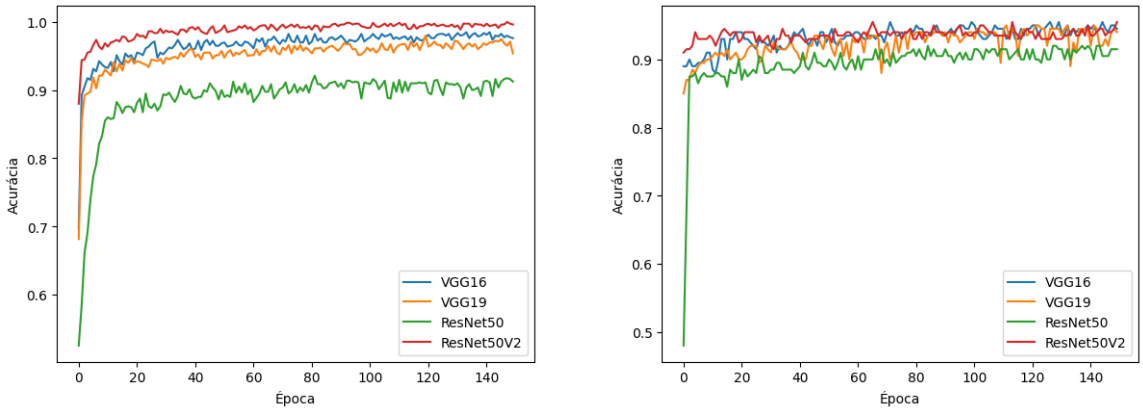
4.1 Resultado dos treinos

Todos os modelos de RNC foram treinados três vezes, com 75, 100 e 150 épocas, permitindo assim, uma análise mais precisa da variação do desempenho ao longo do tempo. A fim de avaliar qual número de épocas obteve o melhor desempenho, o parâmetro utilizado foi a acurácia de treino e validação. A Tabela 3 apresenta as médias de acurácia de treino e validação.

Modelo	**Número de épocas**	**Acurácia de treino média**	**Acurácia de validação média**
VGG16	75	95,32%	92,55%
	100	96,00%	93,10%
	150	96,58%	93,36%
VGG19	75	93,79%	91,26%
	100	94,39%	91,92%
	150	95,20%	92,36%
ResNet50	75	86,13%	85,91%
	100	87,66%	86,63%
	150	88,75%	87,48%
ResNet50V2	75	98,09%	93,08%
	100	98,54%	93,32%
	150	98,81%	93,59%

Tabela 3 – Valores de acurácia média de treino e validação.

Ao analisar a média de acurácia de todos os modelos de RNC, pode-se constatar que o treinamento com 150 épocas obteve maior eficiência. Sendo assim, para os demais procedimentos deste trabalho, foram utilizados os modelos treinados com 150 épocas. A Figura 12 apresenta, por meio de gráficos de acurácia por época, um detalhamento mais preciso do desempenho de treino e validação dos modelos treinados com 150 épocas.

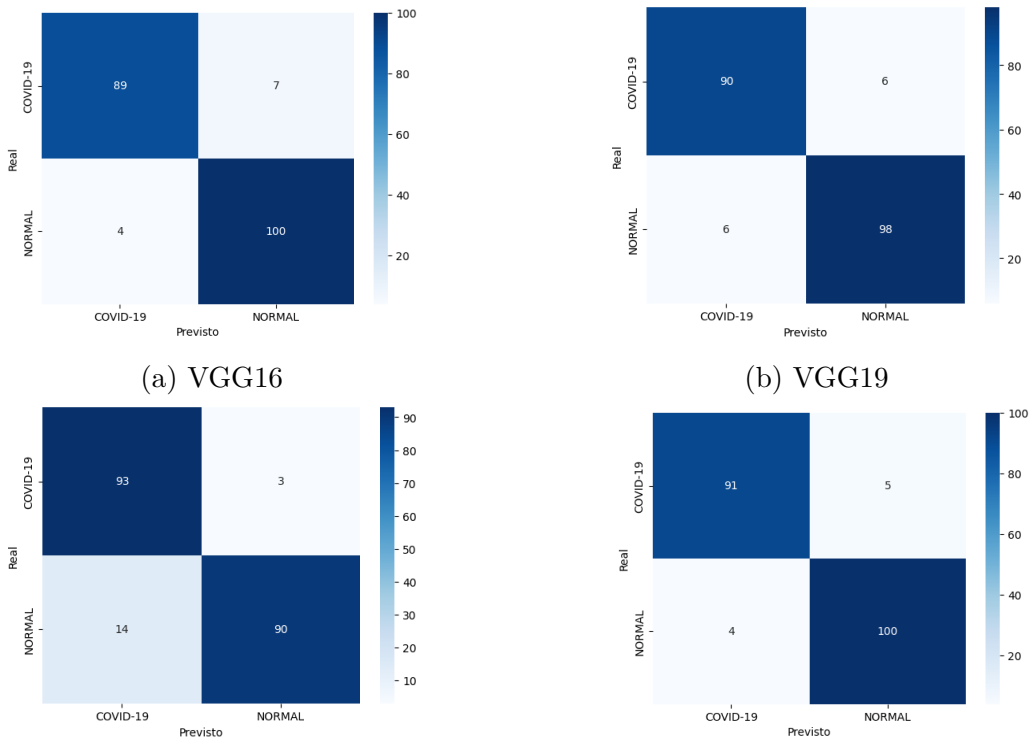


(a) Acurácia do treino. (b) Acurácia de validação.

Figura 12 – Acurácia do treino e validação dos modelos treinados com 150 épocas.

4.2 Resultado dos testes

O desempenho dos modelos de RNC foi medido por meio do processo de teste, o qual recebeu como entrada 200 imagens - 104 da classe Normal e 96 da classe Covid - e realizou a predição da classe à qual cada imagem pertence. Após todas as imagens terem sido testadas, o processo retornou a matriz de confusão de cada modelo, as quais estão representadas na Figura 13.



(a) VGG16 (b) VGG19

(c) ResNet50 (d) ResNet50V2

Figura 13 – Matrizes de confusão dos modelos.

Com a matriz de confusão em mãos, é possível analisar a quantidade de acertos do modelo, somando os valores da diagonal principal, e também a quantidade de erros, somando os valores da diagonal secundária. Com isso, seguindo as equações disponíveis na Seção 2.6, é possível calcular as métricas de avaliação, tais como acurácia, precisão, revocação e pontuação F1. A Tabela 4 apresenta os valores de acurácia de cada modelo no processo de teste, representando a porcentagem de acertos do modelo no conjunto de teste.

A Tabela 5 apresenta os valores das demais métricas de avaliação. Essas métricas são calculadas para as duas classes de imagens no conjunto de teste. A porcentagem de precisão indica o quão corretamente a rede neural é capaz de classificar os dados. Já a porcentagem de revocação é uma medida usada para avaliar a capacidade da rede neural em detectar todas as amostras de uma classe específica. Por fim, a Pontuação F1 é uma medida usada para avaliar a qualidade de um modelo de classificação.

Modelo	**Acurácia**
VGG16	94,50%
VGG16	94,00%
ResNet50	91,50%
ResNet50V2	95,55%

Tabela 4 – Valor de acurácia dos modelos treinados.

Modelo	**Classe**	**Precisão**	**Revocação**	**Pontuação F1**
VGG16	Covid	96%	93%	94%
	Normal	93%	96%	95%
VGG19	Covid	94%	94%	94%
	Normal	94%	94%	94%
ResNet50	Covid	87%	97%	92%
	Normal	97%	87%	91%
ResNet50V2	Covid	96%	95%	95%
	Normal	95%	96%	96%

Tabela 5 – Valores de precisão, revocação e pontuação F1 dos modelos treinados.

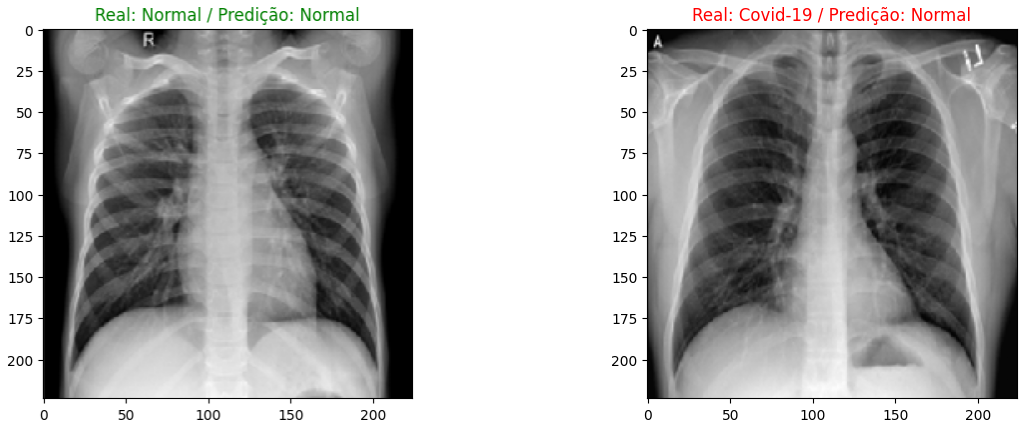
4.3 Análise dos resultados

A análise de desempenho dos modelos de RNC levou em consideração os resultados dos processos de treinamento, validação e teste apresentados nas Seções 4.1 e 4.2. O objetivo da análise é tirar conclusões gerais a partir dos resultados apresentados e chegar a uma conclusão sobre qual dos modelos de RNC implementados é o mais indicado para classificar as imagens de exames de raio-X do tórax do conjunto de dados utilizado.

Durante o processo de treinamento e validação, o modelo ResNet50V2 alcançou a convergência mais rápida e obteve a acurácia média mais alta. Possivelmente, a utilização

de blocos residuais na arquitetura possibilitou que a rede não perdesse informações ao longo das camadas, permitindo um treinamento mais eficiente e preciso.

Assim como no processo de treinamento, o modelo ResNet50V2 obteve a melhor acurácia durante o processo de teste, entre os modelos implementados. Ele acertou a predição de 191 das 200 imagens testadas, alcançando uma acurácia de 95,55% e uma pontuação F1 de 95% e 96% para as classes Covid e Normal, respectivamente. O modelo apresentou como erro mais frequente a predição de uma imagem como Normal, quando na verdade, a imagem pertencia à classe Covid, caracterizando o que é conhecido como falso-Normal. A Figura 14 apresenta dois exemplos de predições do modelo ResNet50V2.

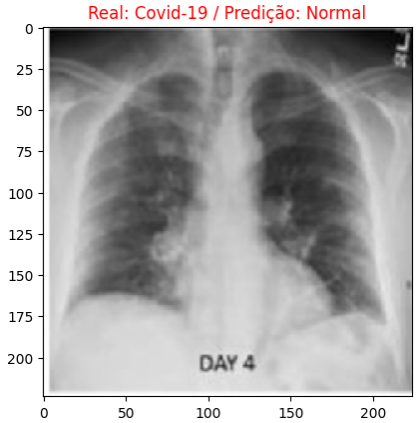


(a) Predição correta. (b) Predição errada.

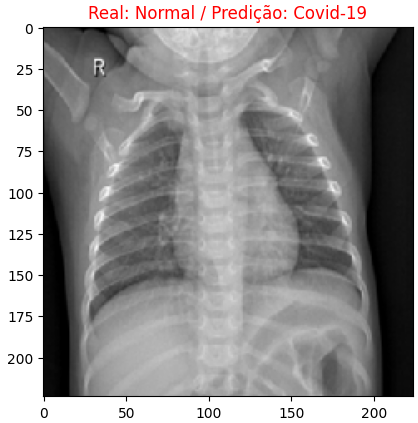
Figura 14 – Exemplos de predições do modelo ResNet50V2.

Os modelos VGG16 e VGG19 obtiveram resultados muito semelhantes durante todo o processo. Durante o teste, os modelos alcançaram uma acurácia de 94,50% e 94%, respectivamente. No entanto, ao avaliar as métricas, o modelo VGG16 apresentou um desempenho ligeiramente superior. É possível observar que a maior diferença entre os resultados dos modelos é o tipo de erro mais comum. Enquanto o modelo VGG16 obteve maior número de erros na predição da classe Normal, o VGG19 apresentou mais erros na predição da classe Covid. A Figura 15 apresenta os erros mais comuns dos modelos.

Uma possível explicação para a ligeira vantagem do modelo VGG16 em comparação com o VGG19 pode ser uma melhor adaptação do modelo ao conjunto de dados utilizado neste trabalho. Além disso, outra possível explicação é que o modelo VGG19 pode ter sofrido mais sobreajuste (*overfitting*) nos dados de treinamento, resultando em um desempenho inferior nos processos de validação e teste.



(a) Predição incorreta do modelo VGG16.



(b) Predição incorreta do modelo VGG19.

Figura 15 – Exemplo de predições incorretas dos modelos VGG16 e VGG19.

O modelo ResNet50 atingiu o pior valor de acurácia dentre as redes implementadas, acertou a previsão de 183 das 200 imagens testadas, atingindo uma acurácia de 91,50%. Analisando os valores das métricas de avaliação, é possível constatar que o modelo apresentou um grande desequilíbrio entre as classes. Dentre os modelos implementados, o ResNet50 obteve a maior quantidade de erros do tipo falso-Covid, porém, também apresentou a menor quantidade de erros do tipo falso-Normal, apenas 3. A Figura 16 apresenta uma das três imagens pelas quais o modelo ResNet50 classificou de forma incorreta como sendo da classe Covid.

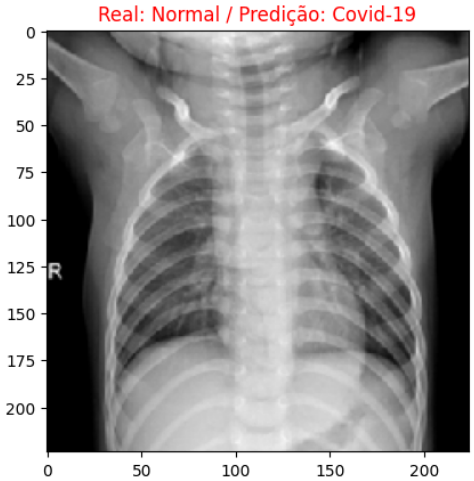


Figura 16 – Predição incorreta do modelo ResNet50.

Com base nos resultados obtidos e analisados utilizando as imagens disponíveis no conjunto de dados e com os hiperparâmetros previamente configurados, verificou-se que, analisando apenas a acurácia, o modelo ResNet50V2 é o mais indicado para realizar a classificação. Durante o processo de treinamento, o modelo atingiu a convergência mais rápida e obteve a melhor pontuação F1 e acurácia durante o processo de teste.

Entretanto, tratando-se de um problema que envolve o diagnóstico de uma doença, a análise de desempenho dos modelos não deve levar em consideração apenas a acurácia. É preciso analisar também os dois tipos de erro. Em uma situação real, um erro falso-Normal pode ter consequências mais graves em comparação ao erro falso-Covid. Um paciente que está doente, mas obteve um exame negativo, pode não ser tratado corretamente e ter seu estado de saúde piorado com o passar do tempo. Por outro lado, um paciente saudável que recebeu um resultado positivo no exame pode passar por uma avaliação médica mais detalhada, o que pode incluir exames auxiliares para o diagnóstico e possibilitar a correção deste erro.

Essa análise pode ser realizada por meio do valor de revocação de ambas as classes de cada modelo, que se refere à capacidade do modelo em detectar corretamente exemplos de imagens de cada classe. Ou seja, quanto maior a revocação da classe Normal, menor será a ocorrência de erros do tipo falso-Covid. E quanto maior a revocação da classe Covid, menor será a ocorrência de erros do tipo falso-Normal.

Nesse sentido, pode-se considerar que o modelo ResNet50 é mais indicado para realizar a classificação, mesmo tendo atingido a acurácia mais baixa entre os modelos implementados. Isso se deve ao fato do modelo ter obtido a maior porcentagem de revocação da classe Covid (97%), o que diminui a chance de ocorrência de erros do tipo falso-Normal, considerado o erro mais grave em um cenário real.

Por outro lado, pode-se afirmar que o modelo menos confiável para realizar a classificação é o VGG16. Embora tenha obtido boas porcentagens de acurácia e pontuação F1, o modelo apresentou a pior revocação da classe Covid (93%) e, consequentemente, registrou a maior quantidade de erros do tipo falso-Normal, totalizando 7.

5 CONCLUSÃO

Este trabalho buscou realizar uma análise comparativa do desempenho dos modelos de redes neurais VGG16, VGG19, ResNet50 e ResNet50V2 na detecção de possíveis diagnósticos positivos de COVID-19 em imagens de raio-X do tórax. Para isso, foi selecionado um conjunto de imagens, sendo uma parte destinada ao treinamento e outra parte utilizada para realizar os testes.

Ao analisar apenas a acurácia, observou-se que o modelo com melhor desempenho é o ResNet50V2, que atingiu uma acurácia de 95,55%. Os modelos VGG16 e VGG19 obtiveram resultados muito similares, com acurácia de 94,50% e 94%, respectivamente. Por fim, o modelo ResNet50 obteve os piores resultados, com 91,50% de acurácia.

Entretanto, levando em consideração a gravidade de um erro falso-Normal, constatou-se que o modelo mais indicado para realizar a classificação é o ResNet50. O modelo obteve 97% de revocação da classe Covid e apresentou apenas três erros do tipo falso-Normal. Da mesma forma, foi constatado que o modelo VGG16 é o menos indicado para realizar a classificação, pois registrou a pior revocação da classe Covid (93%) e obteve sete erros do tipo falso-Normal.

Como estudos futuros, podem ser propostos novos modelos de redes neurais para realizar a mesma classificação, com o objetivo de comparar os resultados do novo modelo com os resultados das redes pré-treinadas utilizadas neste trabalho. Além disso, podem ser realizados testes em conjuntos de dados maiores para verificar se os resultados obtidos neste trabalho se mantêm. Por fim, pode-se realizar análises mais detalhadas sobre as características das imagens de raio-x que influenciam no desempenho dos modelos de redes neurais.

REFERÊNCIAS

[1] COVID-19 Radiography Database. 2022. <https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database>. Acessado em: 11/04/2023.

[2] ANDRADE ALBUQUERQUE, M. P. D. Processamento digital de imagens. *Centro Brasileiro de Pesquisas Fisicas*, 2003.

[3] O'SHEA, K.; NASH, R. An introduction to convolutional neural networks. *CoRR*, abs/1511.08458, 2015. Disponível em: <http://arxiv.org/abs/1511.08458>.

[4] SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[5] HE, K. et al. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. Disponível em: <http://arxiv.org/abs/1512.03385>.

[6] HE, K. et al. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016. p. 770–778.

[7] KERAS website. 2023. <https://keras.io/>. Acessado em: 17/04/2023.

[8] CORONAVIRUS disease (COVID-19) pandemic. 2019. <https://www.who.int/emergencies/diseases/novel-coronavirus-2019>. Acessado em: 12/01/2023.

[9] CARVALHO Élisson de et al. Automatic detection of covid-19 in x-ray images using fully-connected neural networks. In: *Anais do XVI Workshop de Visão Computacional*. Porto Alegre, RS, Brasil: SBC, 2020. p. 41–45. ISSN 0000-0000. Disponível em: <https://sol.sbc.org.br/index.php/wvc/article/view/13478>.

[10] SALEHI, S. Coronavirus disease 2019 (COVID-19): A systematic review of imaging findings in 919 patients. " AJR. *AJR. American journal of roentgenology*, v. 215, p. 87–93, 2020.

[11] FLECK, L. et al. Redes neurais artificiais: Princípios básicos. *Revista Eletrônica Científica Inovação e Tecnologia*, v. 1, n. 13, p. 47–57, 2016.

[12] CIOTTI, M. et al. The covid-19 pandemic. *Critical Reviews in Clinical Laboratory Sciences*, Taylor and Francis, v. 57, n. 6, p. 365–388, 2020. PMID: 32645276. Disponível em: <https://doi.org/10.1080/10408363.2020.1783198>.

[13] TECUCI, G. Artificial intelligence. *WIREs Computational Statistics*, v. 4, n. 2, p. 168–180, 2012. Disponível em: <https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.200>.

[14] WANG, S.-C. Artificial neural network. In: _____. *Interdisciplinary Computing in Java Programming*. Boston, MA: Springer US, 2003. p. 81–100. ISBN 978-1-4615-0377-4. Disponível em: <https://doi.org/10.1007/978-1-4615-0377-4_5>.

[15] LEITE, D. B. e Mayra Binoti e H. ConfiguraÇÃo de redes neurais artificiais para estimaÇÃo do volume de Árvores. *Revista Ciência da Madeira (Brazilian Journal of Wood Science)*, v. 5, n. 1, p. 10–12953/2177–6830.v05n01a06, 2014. ISSN 2177-6830. Disponível em: <https://periodicos.ufpel.edu.br/ojs2/index.php/cienciadamadeira/article/view/4067>.

[16] JMOUR, N.; ZAYEN, S.; ABDELKRIM, A. Convolutional neural networks for image classification. In: *2018 International Conference on Advanced Systems and Electric Technologies* (IC_ASET). [S.l.: s.n.], 2018. p. 397–402.

[17] BANERJEE, C.; MUKHERJEE, T.; PASILIAO, E. An empirical study on generalizations of the relu activation function. In: *Proceedings of the 2019 ACM Southeast Conference*. New York, NY, USA: Association for Computing Machinery, 2019. (ACM SE '19), p. 164–167. ISBN 9781450362511. Disponível em: <https://doi.org/10.1145/3299815.3314450>.

[18] MAAS, A. L. Rectifier nonlinearities improve neural network acoustic models. In: . [S.l.: s.n.], 2013.

[19] SHARMA, S.; SHARMA, S.; ATHAIYA, A. Activation functions in neural networks. *International Journal of Engineering Applied Sciences and Technology*, v. 04, p. 310–316, 05 2020.

[20] THECKEDATH, D.; SEDAMKAR, R. R. Detecting affect states using vgg16, resnet50 and se-resnet50 networks. *SN Computer Science*, v. 1, 2020.

[21] COLAB website. 2023. <https://colab.research.google.com/notebooks/intro.ipynb>. Acessado em: 17/04/2023.

[22] IMAGENET website. 2023. <https://www.image-net.org/>. Acessado em: 17/04/2023.

[23] YING, X. An overview of overfitting and its solutions. *Journal of Physics: Conference Series*, v. 1168, p. 022022, 02 2019.