UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

STHEFANIE MONICA PREMEBIDA

THIAGO FELLIPE ORTIZ DE CAMARGO

APRENDIZADO PROFUNDO PARA AUXILIAR A DETECÇÃO DE COVID-19 BASEADO EM IMAGENS DE RAIO-X DE TÓRAX : UMA ABORDAGEM PRÁTICA

PONTA GROSSA

2021

STHEFANIE MONICA PREMEBIDA

THIAGO FELLIPE ORTIZ DE CAMARGO

APRENDIZADO PROFUNDO PARA AUXILIAR A DETECÇÃO DE COVID-19 BASEADO EM IMAGENS DE RAIO-X DE TÓRAX : UMA ABORDAGEM PRÁTICA

DEEP LEARNING TO ASSIST COVID-19 DETECTION BASED ON CHEST X-RAY IMAGES : PRACTICAL METHODOLOGY

Trabalho de Conclusão de Curso apresentado como requisito para obtenção do título de Bacharela/Bacharel em Engenharia Elétrica da Universidade Tecnológica Federal do Paraná (UTFPR).

Orientadora: Profa. Dra. Marcella Scoczynski Martins
Coorientadora: Profa. Dra. Cristhiane Goncalves

PONTA GROSSA

2021

STHEFANIE MONICA PREMEBIDA

E

THIAGO FELLIPE ORTIZ DE CAMARGO

APRENDIZADO PROFUNDO PARA AUXILIAR A DETECÇÃO DE COVID-19 BASEADO EM IMAGENS DE RAIO-X DE TÓRAX: UMA ABORDAGEM PRÁTICA

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Bacharela/Bacharel em Engenharia Elétrica da Universidade Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: 09 de Dezembro de 2021

_____

Prof.(a) Dra. Marcella Scoczynski Ribeiro Martins
Universidade Tecnológica Federal do Paraná

_____

Prof.(a) Dra. Cristhiane Goncalves
Universidade Tecnológica Federal do Paraná

_____

Prof.(a) Dra. Fernanda Cristina Correa
Universidade Tecnológica Federal do Paraná

_____

Prof. Dr. Marcio Rodrigues da Cunha Reis
Instituto Federal De Goiás

_____

Prof. Dr. Antonio Vanderley Herrero Sola
Universidade Tecnológica Federal do Paraná

PONTA GROSSA

2021

’Em um estado sombrio nós nos encontramos, um pouco mais de conhecimento iluminar nosso caminho pode’ - Mestre Yoda

AGRADECIMENTOS

Todo nosso crescimento durante todo esse período da graduação, seja pessoal, acadêmico ou profissional, não teria sido possível sem diversas pessoas que estavam ao nosso lado.

Este trabalho foi mais um dos trabalhos da graduação desenvolvidos por uma dupla de amigos que acompanham a jornada um do outro à 7 anos, a amizade e companheirismo entre nós foi imprescindível durante toda a jornada.

Gostaríamos de agradecer nossas famílias que sempre nos deram suporte, amor, carinho e se mostravam abertas a apoiar nossos sonhos.

Agradecemos aos nossos professores que foram como mestres durante nosso processo de crescimento intelectual, nos mostraram diversos caminhos e sempre nos davam a oportunidade de aprender mais.

Aos nossos amigos, que nos ajudaram direta e indiretamente nos nossos melhores e piores momentos e que sempre estiveram conosco nas adversidades que surgiram durante o curso.

Os autores são gratos pelo auxilio do profissional da medicina Mayler Olombrada, que disponibilizou seu tempo e conhecimento para guiar nosso percusso no preparo das imagens de raio-X de tórax.

E o nosso agradecimento especial vai para a orientadora deste trabalho, Prof. Marcella. Sem você, nós não teríamos metade da jornada acadêmica que temos hoje e você sempre vê nossa melhor versão acadêmica e como pessoas, somos infinitamente gratos por todo cuidado e ensinamentos que trouxe para nós durante esses anos. Você é uma profissional excelente e uma pessoal incrível, esperamos que um dia sejamos pelo menos metade de como você é para nós.

RESUMO

Dado o grande número de casos COVID-19 em todo o mundo, uma solução prática para reduzir e aliviar a fila de pacientes em hospitais e sistemas de saúde é bem-vinda. Diagnósticos rápidos e confiáveis baseados em ferramentas tecnológicas podem ajudar os profissionais da medicina a administrar essa situação de gargalo. Neste trabalho, propomos uma metodologia prática usando aprendizado profundo (*deep learning*) para detectar e classificar os pulmões afetados por COVID-19 por meio de imagens de raio-x de tórax. A arquitetura RetinaNet é utilizada no processo. Esta arquitetura é um detector de um estágio combinado com a *focal loss*. Consideramos um conjunto de dados com 5000 imagens, sendo 2000 para treinamento, 1000 para validação e 2000 para teste. Os resultados obtidos mostram um escore de especificidade de 0.99, precisão de 0.99, sensitividade de 0.56 e mAP de 0.81. A alta pontuação de especificidade implica que um paciente com COVID-19 tem alta probabilidade de ser diagnosticado corretamente.

Palavras-chave: Apredizado Profundo; Imagens de Raio-X de Tórax; Detecção de COVID-19.

ABSTRACT

Given the large number of COVID-19 cases around the world, a practical solution to reduce and alleviate the patient queue in hospitals and healthcare systems is welcome. Fast and reliable diagnostics based on technological tools can help medical professionals to manage this bottleneck situation. In this work, we propose a practical methodology using deep learning to detect and classify lungs affected by COVID-19 through chest x-ray imaging. RetinaNet architecture is used in the process. This architecture is a one-stage detector combined with Focal Loss. We considered a dataset with 5000 images, 2000 to train the model, 1000 to validate, and 2000 to test the model. The results obtained show a recall score of 0.99, precision of 0.99, sensibility of 0.56, and mAP of 0.81. The high recall score implicates that a patient with COVID-19 will be classified correctly.

Keywords: Deep Learning; Chest X-ray Imaging; COVID-19 Detection.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

SIGLAS

AP	Precisão média, do Inglês *Average Precision*
CNN	Rede Neural Convolucional, do Inglês *Convolutional Neural Network*
CXR	Raio-X de Tórax, do inglês *Chest X-ray*
Fast R-CNN	Rede convolucional rápida baseada em região, do Inglês *Fast Region-based Convolutional Network*
FN	Falso Negativo, do Inglês *False Negative*
FP	Falso Positivo, do Inglês *False Positive*
FPN	Rede Piramidal de Características, do Inglês*Feature Pyramid Network*
GPU	Unidades de Processamento Gráfico, do Inglês *Graphics Processing Unit*
IoU	Intersecção sobre a união, do Inglês *Intersection over Union*
LUS	Ultrassom de Pulmão, do inglês *Lung Ultrassound*
mAP	Precisão média média, do Inglês *Mean Average Precision*
MLP	Perceptron de múltiplas camadas, do Inglês*Multilayer Perceptron*
PNG	Gráficos Portáteis de Rede, do Inglês *Portable Network Graphics*
R-CNN	Rede Neural Convolucional Baseada em Região, do inglês*Region-Based Convolutional Neural Network*
ReLU	Unidade Linear Retificada *Rectified Linear Unit*
RNA	Redes Neurais Artificiais *Artifficial Neural Networks*
RoI	Região de interesse, do Inglês *Region of Interest*
SGD	Gradiente Descendente Estocástico, do Inglês *Stochastic Gradient Descent*
SSD	Detector de Disparo Único, do Inglês *Single Shot Detector*
SVM	Máquina de vetores de suporte, do Inglês *Support Vector Machine*
TC	Tomografia Computadorizada
TN	Verdadeiro Negativo, do Inglês *True Negative*
TP	Verdadeiro Positivo, do Inglês *True Positive*
UTFPR	Universidade Tecnológica Federal do Paraná
XML	Linguagem Padronizada de Marcação Genérica, do inglês*Extensible Markup Language*
YOLO	Você só olha uma vez - tipo de detector de disparo único, do Inglês *You Only Look Once*

SUMÁRIO

1 INTRODUÇÃO

Doenças infecciosas sempre assolaram a humanidade. No passado, normalmente, a causa destas pestes e epidemias eram desconhecidas, entretanto, com o progresso de várias áreas da ciência as causas e métodos de prevenção foram aprendidas e disseminadas pelo mundo (MARTINS *et al.*, 1997).

No ano de 2019 a humanidade começou a enfrentar o vírus COVID-19, que causa infecção no trato respiratório superior e nos pulmões (ISMAEL; ŞENGÜR, 2021). O vírus se espalhou rapidamente acarretando uma situação de pandemia (TIAN *et al.*, 2020).

As técnicas de imagem para identificação de COVID-19 mais comuns são tomografia computadorizada (TC), ultrassom dos pulmões (LUS), e raio-X de tórax (CXR) (KHUZANI; HEIDARI; SHARIATI, 2021). Dentre estes, o exame de raio-X provém uma análise rápida e acessível, além de ter baixo custo e infligir uma dose pequena de radiação (ROY *et al.*, 2020; WANG, S.; ZHA *et al.*, 2020). Sendo assim, raio-X de tórax foi aplicado como a primeira técnica de imagem para auxiliar o diagnóstico de vários tipos de doenças, bem como para auxiliar na inferência da severidade de doenças, guiando a decisão de prioridade de tratamento. Este fato contribui para minimizar a saturação dos sistemas de saúde durante a pandemia de COVID-19 (OH; PARK; YE, 2020).

Desta forma, a aplicação de técnicas de inteligência artificial, especialmente baseados em aprendizado de máquina, podem ser uma ferramenta útil na gestão de pacientes suspeitos de infecção pelo novo coronavirus. Técnicas baseadas em aprendizado de máquina apresentam implementações escaláveis por meio do processamento de imagem para classificação, detecção e segmentação. A classificação é realizada com base nas características relevantes extraídas da imagem automaticamente (SUN *et al.*, 2019).

A técnica mais utilizada em aprendizado de máquina para processamento de imagem são as redes neurais convolucionais. Este tipo de rede consiste, de forma simplista, em um conjunto de camadas de convolução (operação de multiplicação matricial entre uma matriz de entrada e filtros com elementos ajustáveis) e *maxpooling* (seleção do elemento de maior valor dentro de um determinado subespaço da matriz original) seguido por uma, ou múltiplas, camadas densas de neurônios totalmente conectados

(KRIZHEVSKY; SUTSKEVER; HINTON, 2012). A primeira etapa é chamada de extração de características, já a segunda de etapa de classificação. Vários modelos de convolução já foram desenvolvidos, atualmente muitas das arquiteturas que circundam o pódio do estado-da-arte na competição *ImageNet Large Scale Visual Recognition Challenge* (RUSSAKOVSKY *et al.*, 2015) são baseadas na *EfficientNet* (TAN; LE, 2019)

A tarefa de classificação pode seguir diversas filosofias: classificação de imagem, em que se pode atribuir uma ou múltiplas classes para a mesma imagem; classificação de objeto, em que o objeto alvo é detectado e classificado, possibilitando uma única imagem conter múltiplos objetos de inúmeras classes.

A tarefa de detecção de objeto segue duas linhas: dois estágios e um estágio. A abordagem de dois estágios carrega esse nome por separar o processo de proposta da localização de objeto do processo de classificação, ao passo que os detectores de um estágio não geram propostas de localização, ao invés disto, performam a detecção ao mesmo tempo que a classificação é realizada sobre vários mapas de características que potencialmente contêm um objeto. A família de detectores de objetos denominadas *Region Based Convolutional Neural Network* (RCNN) (REN *et al.*, 2015) é um exemplo de detector de dois estágios. Já a *RetinaNet*(LIN; GOYAL *et al.*, 2017) e a *Single Shot MultiBox Detector* (LIU, W. *et al.*, 2016) são exemplos de detectores de um estágio.

Entretanto, esta abordagem tem grande contraponto, visto que o fundo da imagem também pode ser considerado como uma classe. Este fato tem como consequência o desbalanceamento entre classes, pois, em grande parte das imagens, o fundo é predominante em relação aos objetos, o que leva ao aprendizado não ótimo do modelo, sendo predominantemente treinado para detectar e classificar a classe de fundo. O problema abordado é chamado de *foreground-background imbalance* (CHEN, J. *et al.*, 2020).

Com esta situação problema em vista, surge a arquitetura *RetinaNet* (LIN; GOYAL *et al.*, 2017) para amenizar o desbalanço entre classes. A *RetinaNet* conta com uma rede piramidal de características (FPN) (LIN; DOLLÁR *et al.*, 2017) para extrair características de diferentes semânticas, ou seja, de diferentes tamanhos e formas. O ponto crucial desta arquitetura é a função de perda, a *Focal Loss*. Esta função busca amenizar o impacto de exemplares fáceis no processo de aprendizado, ou seja, ameniza os sujeitos que possuem alto coeficiente de probabilidade de conter um objeto, forçando o modelo a focar seu aprendizado em exemplares de difícil detecção. Em outas palavras,

a *Focal Loss* é uma função *crossentropy* dinamicamente escalada, modulada de forma inversamente proporcional pelo coeficiente de confiança dos objetos.

Abordagens que utilizam imagens de raio-X de tórax para classificação de pulmões já foram propostas, como em (BALTRUSCHAT *et al.*, 2019), em que os autores empregam diferentes arquiteturas, diferentes técnicas de pré-processamento de imagem, metadados dos pacientes e técnicas de treinamento como transferência de aprendizado e ajuste fino, a fim de classificar as imagens do banco *ChestX-ray14* em quatorze classes diferentes.

Recentemente, múltiplas metodologias para classificar pulmões com COVID-19 foram propostas, em (HEMDAN; SHOUMAN; KARAR, 2020) os autores apresentam um *ensemble* de sete redes neurais convolucionais para classificar imagens de raio-X de tórax entre positivo ou negativo para COVID-19. Já em (NARIN; KAYA; PAMUK, 2021), os autores utilizam a técnica de transferência de aprendizado em três diferentes arquiteturas a fim de comparação de performance na tarefa de classificação de imagens de raio-X de tórax entre pneumonia bacteriana, COVID-19, Normal e pneumonia viral.

Similar a metodologia proposta neste trabalho, os autores em (OZTURK *et al.*, 2020) desenvolvem uma arquitetura baseada em detecção de objeto, para detectar e classificar o pulmão entre COVID-19, *No-Findings* e Penumonia Viral. Já em (SAIZ; BARANDIARAN, 2020), os autores utilizam a arquitetura de detecção de um estágio *Single-Shot Object Detection - 300* para detectar e classificar cada lobo pulmonar entre as classes normal e COVID-19.

O objetivo principal deste trabalho é propor um fluxo de trabalho prático para detectar e classificar pulmões em imagens de raio-X de tórax através de arquitetura de detecção de um estágio. A arquitetura é baseada na *RetinaNet*, um modelo de aprendizado profundo que emprega a função de perda *Focal Loss* para resolver o problema de desbalanceamento de classes durante a fase de treinamento.

A metodologia proposta possibilita a detecção e classificação do pulmão de forma instantânea e precisa em um conjunto de dados de características variadas dentre as classes Normal e COVID-19. Além disso, o fluxo de trabalho abrange a troca entre tempo de execução e acurácia, de forma a visar a aplicação pratica, o que torna a proposta útil para auxiliar a interpretação certeira dos profissionais da saúde quanto a imagens de raio-x de tórax.

O trabalho segue a seguinte ordem: na seção 2 é apresentado o trajeto do

desenvolvimento do aprendizado de máquina desde a criação do neurônio artificial até arquiteturas complexas para tarefa de detecção de objetos, bem como as métricas para avaliação de performance, e, por último, trabalhos relacionados. A seção 3 explana a elaboração da metodologia proposta, aplicação de ferramentas e organização dos recursos: bancos de imagens. A seção 4 discute os resultados obtidos traçando uma base de comparação entre dois modelos. Por fim, a seção 5 apresenta uma inferência baseada nos resultados obtidos e na complexidade da metodologia, como também apresenta direções futuras.

2 REVISÃO DA LITERATURA

Este capítulo tem finalidade de realizar uma revisão bibliográfica em relação ao COVID-19, redes neurais e outras técnicas que serão aplicadas no decorrer do trabalho.

2.1 COVID-19

A COVID-19 é a infecção causada pelo SARS-Cov-2 e em março de 2020 a Organização Mundial da Saúde decretou estado de pandemia. Como a família de coronavírus é de doenças respiratórias, a COVID-19 muitas vezes pode passar despercebido, já que se assemelha a um resfriado, mas em casos mais graves pode apresentar sintomas de pneumonia. Os principais sintomas clínicos são dificuldade para respirar - fadiga, acompanhado de febre alta, tosse, coriza, mal estar, dor de cabeça, perda do olfato, entre diversos outros sintomas que podem aparecer conforme a gravidade da doença (WORLD..., 2020).

Antes das vacinas serem disponibilizadas para auxiliar na prevenção e na diminuição de número de infectados com a doença, os principais cuidados com pessoas infectadas incluíam a quarentena em casos leves, uso de oxigênio para pacientes em estado grave e ventiladores para os que estão em risco da evolução para doença grave, ou seja, o pulmão é um dos órgãos mais afetados pelo COVID-19. Pacientes em estado grave tem seus alvéolos, estrutura que faz a troca gasosa dos pulmões captando O_2 e liberando CO_2, infectados, o que gera alterações e faz com que essas células morram, desencadeando ainda mais problemas, como o processo de inflamação e edema pulmonar, excesso de líquidos, que impedem a troca gasosa e causam a insuficiência respiratória, segundo a pesquisadora Marisa Dolhnikoff em bbc2021.

Segundo às Diretrizes para Diagnóstico e Tratamento da COVID-19, o diagnóstico da doença pode ser feito primeiramente em casos suspeitos, que são pessoas com Síndrome Gripal, onde apresenta um quadro respiratório agudo, febre ou estado febril, acompanhado de tosse, dor de garganta, coriza ou fadiga (MINISTÉRIO..., 2020). Outro caso de suspeita pode ser quando a pessoa apresenta Síndrome Respiratória Aguda Grave, onde tem desconforto para respirar, ou pressão constante no tórax, ou saturação de O_2 com valor menor que 95%.

Para confirmar um caso de COVID-19, pode ser usado o critério laboratorial,

como o exame RT-PCR, onde a detecção do vírus SARS-CoV-2 é instantânea. Ainda em critério laboratorial, pode ser feito o teste imunológico, sendo o teste rápido ou de sorologia clássica para detectar anticorpos, e o resultado positivo para IgM e/ou IgG representa a que a pessoa está contaminado com o novo coronavírus, porém, esse teste tem sua amostra coletada apenas após o sétimo dia de início dos sintomas. Além dos testes moleculares, temos o uso de exames de imagem, como Raio-X de tórax de pacientes com suspeitas de pneumonia e Tomografia Computadorizada (TC) do tórax de pacientes com acometimento do trato respiratório inferior (MINISTÉRIO..., 2020).

A pandemia afetou os países de diversas formas, políticas, econômicas, culturais e principalmente sociais. O sistemas de saúde entraram em colapso por conta da quantidade de pessoas infectadas e países que estavam com medidas não tão restritivas, como o Brasil, tiveram diversos novos casos por dia e diversos dias que somaram mais de mil mortes. Ao total, apenas no Brasil, desde o dia 26 de fevereiro de 2021 até o dia 12 de novembro de 2021, quase 21,94 milhões de pessoas foram infectadas com o COVID-19 e 610.491 pessoas morreram (CORONAVIRUS..., 2021).

O desemprego aumentou no ano de 2020 em relação à 2019, como por exemplo nos Estados Unidos que em 2019 a taxa de desemprego era de 3,7% e em 2020 cresceu para 8,9%, no Brasil a taxa passou de 11,9% para 13,4% (CORONAVÍRUS:..., 2021). O comércio precisou se adaptar com as vendas online, as empresas com o trabalho remoto e diversas áreas precisaram se reinventar, como o setor de turismo, que foi fortemente afetado durante esse período. O mundo precisou de adaptação da realidade, com novos hábitos e formas diferentes de realizar tarefas, principalmente na hora de diagnosticar o COVID-19 e diversas outras doenças de formas mais rápidas e tecnológicas.

2.2 REDES NEURAIS ARTIFICIAIS

As redes neurais artificiais tiveram seu início na década de 40 com Warren McCulloch e Walter Pitts, que criaram a primeira representação artificial do neurônio utilizando métodos matemáticos (MCCULLOCH; PITTS, 1943). Como um neurônio, a entrada da rede recebe sinais como os dendritos em neurônios biológicos, essa informação é processada pelo corpo e no axônio o sinal é transmitido para os outros neurônios, como ilustrado na Figura 1.

O modelo computacional de um neurônio, Figura 2, é formado pelos sinais de

entradas, representados pelo vetor $x = [x_1, x_2, x_3, \cdots, x_n]$, onde n é o valor total de sinais de entradas usados no neurônio, e assim que chegam ao neurônio são multiplicados pelos pesos sinápticos, vetor $w = [w_1, w_2, w_3, \cdots, w_n]$, gerando um valor Z, chamado de potencial de ativação e o valor de *b* fornece um grau de liberdade, já que não é afetado pela entrada, e corresponde ao viés do modelo. O potencial de ativação (Z) passa então por uma função de ativação σ, que é uma função matemática não linear, e o resultado é um valor y, que chamamos de saída do modelo.

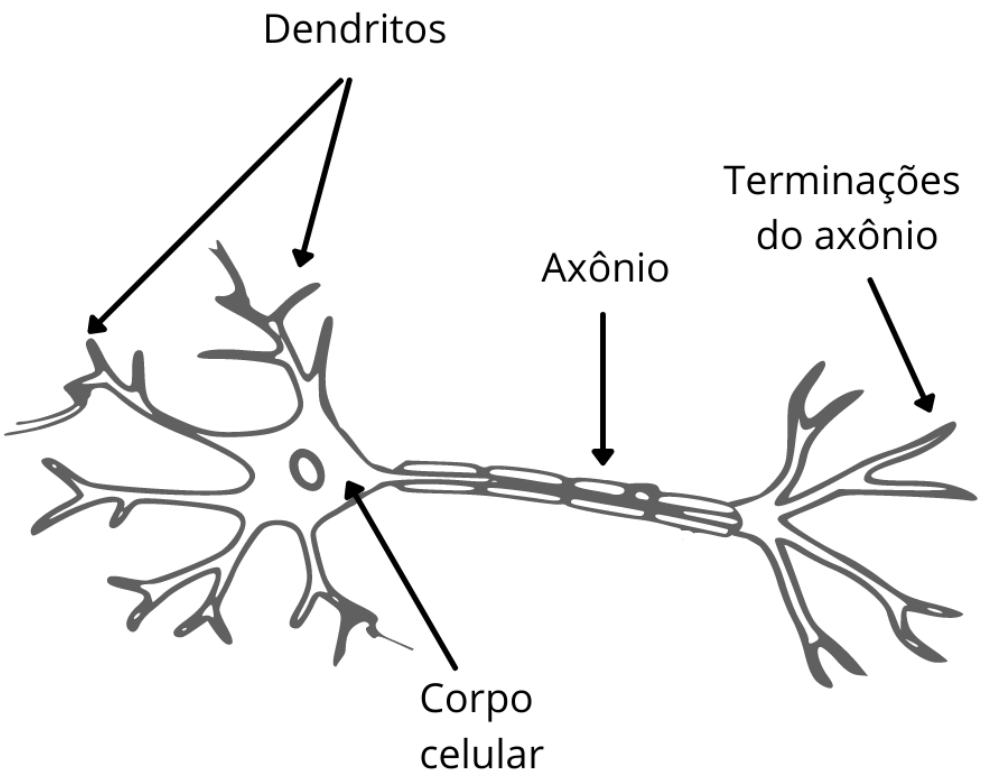


Figura 1 – Neurônio biológico.

Já na década de 50, Frank Rosenblatt criou o Perceptron (ROSENBLATT, 1957). Depois de alguns anos Marvin Minsky e Seymour Papert detectaram uma limitação da rede Perceptron, onde ela só conseguia resolver problemas se a resposta fosse uma

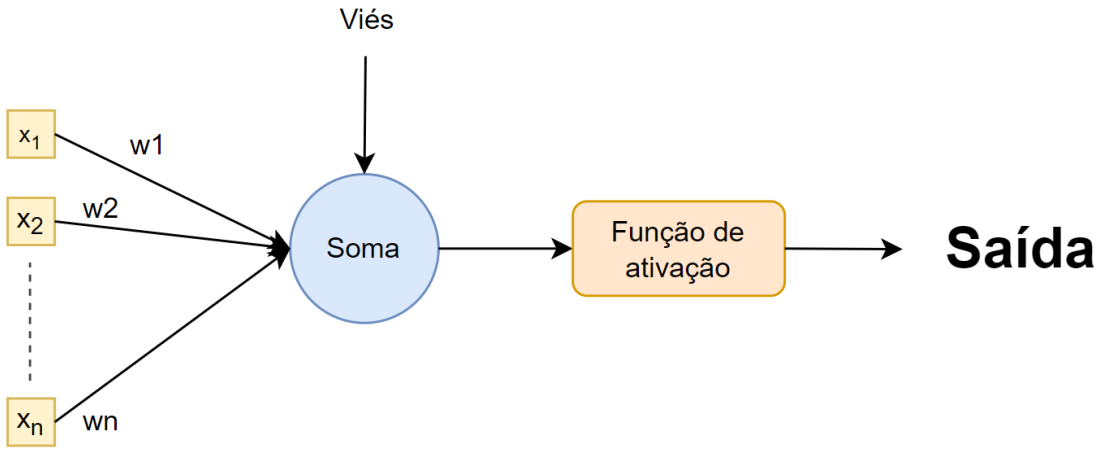


Figura 2 – Neurônio artificial.

função linear (reta), e foi dado o nome de Problema do XOR (ou exclusivo), e necessitava de uma função não-linear para resolver esse problema (MESQUITA; PIMENTA, s.d.). Diante da limitação da rede Perceptron, os avanços sobre redes neurais pararam, até que em 1986 Geoffrey Hinton sugeriu um novo modelo, Multilayer Perceptron (MLP) e trouxe à tona um novo conceito, chamado *Backpropagation*.

A MLP pode conter múltiplas camadas escondidas de neurônios, o que eleva seu potencial de generalização para diversos problemas de diferentes naturezas, ou seja, a MLP possui caráter de generalizador universal. Na Figura 3 a estrutura de uma rede MLP de duas camadas é ilustrada. A camada de entrada ainda tem o vetor $x = [x_1, x_2, x_3, \cdots, x_n]$, e os pesos sinápticos não são representados por uma letra em, específico, mas cada ligação entre a entrada e a primeira camada tem um peso sináptico, assim como as iterações entre camadas, e a camada de saída pode ter um número *n* de saídas possíveis.

O *Backpropagation*, ou retropropagação em português, que nada mais é que, com base no cálculo do erro ocorrido na camada de saída da rede, ela recalcula todos os pesos sinápticos w da rede, indo da última camada até a primeira (de trás para frente), afim de melhorar os resultados finais. Toda rede neural é composta por extensas combinações e fórmulas matemáticas que combinadas geram o que conhecemos como o algoritmo, que podem ser arranjadas de *n* formas até o melhor resultado final, e a retropropagação faz o papel de reorganizar alguns dos valores dos pesos para que esse resultado seja o mais próximo do esperado. Na seção 2.2.1, o gradiente descendente e

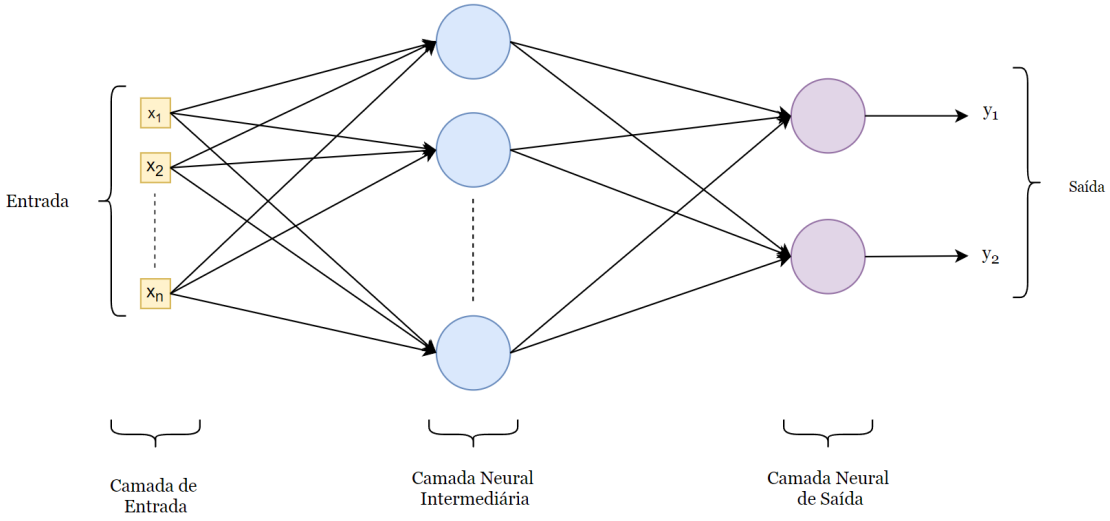


Figura 3 – Representação de uma rede MLP

algoritmos de otimização, que são responsáveis pela retropropagação serão discutidos com mais detalhes..

Todos esses esforços, humano e computacional, precisam ser alocados em algum lugar e resolver problemas do mundo real, não apenas problemas teóricos e lógicos, como os primeiros sugeridos que envolviam jogos como o xadrez. Cada vez mais as Redes Neurais Artificiais (RNA) estão sendo utilizadas em diversos campos do mundo, desde aplicações básicas de utilização para classificação de spam (SILVA, 2009), aplicações na saúde como a predição de doenças como a Hepatite A (SANTOS *et al.*, 2005), em análises de créditos em bancos(PIRES *et al.*, 2008) ou até no auxílio de detecção e classificação de defeitos em linhas de transmissão (JUNGES, 2018). *Chatbots*, análises de sentimentos, gerenciamento de informações, análise de séries temporais, entre outros possíveis temas estão sendo explorados cada vez mais, porém um dos campos onde as RNAs não conseguem bons resultados são os trabalhos diretos com imagens.

Quando surgem problemas que envolvem imagens e precisam ser resolvidos com redes neurais, é necessário a utilização de redes com uma camada de convolução que aplica um filtro (*kernel*), e essa camada serve para aprender padrões nas imagens, como arestas, cantos, etc.. Redes neurais que contém uma camada de convolução são nomeadas de Redes Neurais Convolucionais ou em inglês *Convolutional Neural Network* (CNN) (ALBAWI; MOHAMMED; AL-ZAWI, 2017).

2.2.1 REDES NEURAIS CONVOLUCIONAIS

As Redes Neurais Convolucionais (CNN) são um tipo de expansão das redes neurais artificiais tradicionais, onde trazem camadas ocultas totalmente conectadas e também camadas convolucionais(PARKHI; VEDALDI; ZISSERMAN, 2015). O que difere uma CNN, além das camadas a mais, é que ela faz a extração de características da imagem e converte em dimensões cada vez menores sem perder suas características principais, como pode ser vista na Fig. 4.

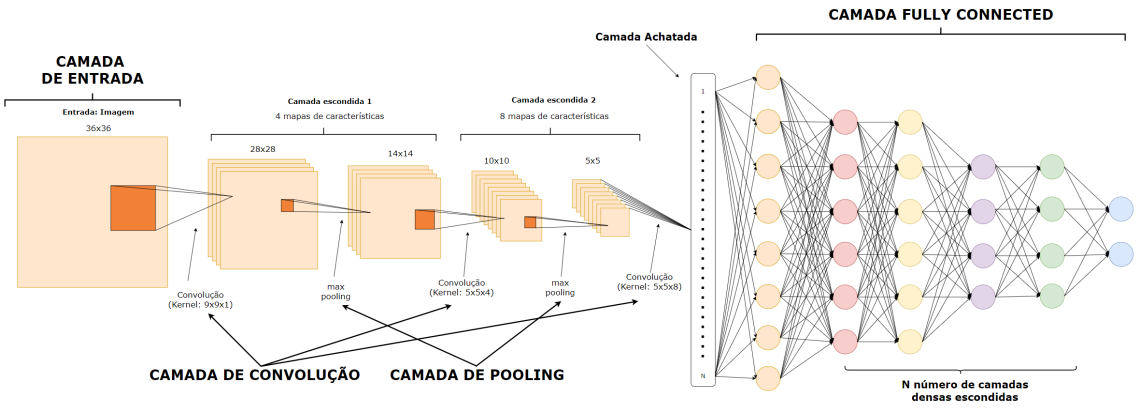


Figura 4 – Arquitetura de uma Rede Neural Convolucional

As redes precisam ter essa característica tridimensional pois quando utiliza-se de imagens, tem-se o plano 2D e mais um eixo, que representa as cores da imagens, e é nesse caso em que a utilização de tensores se faz útil, já que as imagens são codificadas em canais de cores (RGB - *Red, Green, Blue*) e a informação que é retirada da imagem é a intensidade das cores em cada canal na largura e altura, então para a cor vermelha é gerada uma matriz de intensidade e o mesmo ocorre para as outras duas cores, verde e azul, e quando combinadas todas essas matrizes formam o **tensor**. A CNN é composta de quatro tipos de camadas diferentes, sem contar a camada de saída, que são listadas abaixo e podem ser verificadas nas divisões da Figura 4.

- Camada de entrada.
- Camada convolucional.
- Camada de *pooling*.
- Camada *fully connected* (FC).

A camada de entrada é onde a imagem é inserida e transformada uma matriz tridimensional, como já foi explicado anteriormente, mas é necessária a reformulação

em coluna única, então se a entrada é uma imagem de dimensões 28x28, a conversão se da em uma matriz 784x1, onde o 748 deriva da multiplicação de 28 por 28.

A camada convolucional é onde começa o trabalho da rede e pode ser chamada de camada extratora, já que extrai os recursos da imagem. A entrada da camada é uma matriz $[h1 * w1 * d1]$, que é multiplicada por **kernels**, que são como filtros. Estes kernels são matrizes de dimensões $[h2 * w2 * d1]$ e são empilhados em um número arbitrário de camadas. Cada kernel é responsável por aprender a extrair uma determinada característica da imagem.

O produto matriz de entrada e kernel é obtido através do 'deslizamento' do kernel sobre a matriz de entrada, obtendo-se uma nova matriz para cada kernel estipulado. Depois de passar por este processo, a saída da camada convolucional tem dimensão $[h3 * w3 * d2]$, e todos esses passos podem ser visualizados na Figura 5.

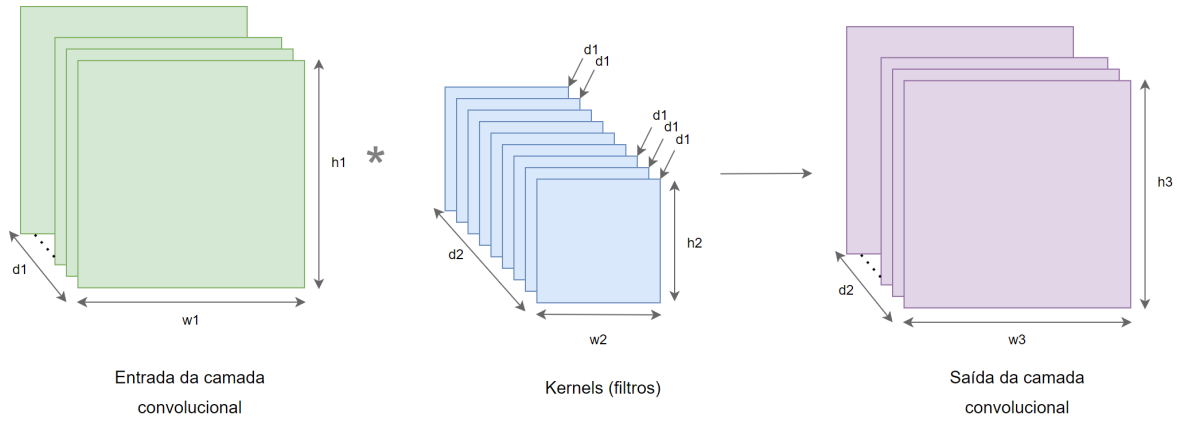


Figura 5 – Camada convolucional de uma CNN.

Ainda na camada convolucional, tem-se mais dois termos importantes no comportamento do modelo, o **Stride** e o **Padding**.

- **Stride:** O valor de stride dita o tamanho do passo que o kernel dá ao deslizar pela matriz de entrada. O kernel se move 1 pixel por vez sobre a matriz, portanto ele tem passo de 1, mas esse valor de passo pode aumentar, por exemplo, para 2, e ele moverá 2 pixels por vez, o que afeta as dimensões do tensor de saída.
- **Padding:** *Padding*, ou preenchimento, acontece quando as matrizes de entrada tem suas bordas preenchidas com zeros a fim de se manter o tamanho original da matriz de entrada na saída do processo de convolução.

Portanto a representação matemática da camada de convolução se dá pelo

valor da característica de localização (i,j), referente ao k-ésimo mapa de características, que é a saída resultante da camada de convolução que ressalta algumas características da imagem como o contorno do objeto, da l-ésima camada, $z^l_{i,j,k}$, e é calculado com base na equação 1, em que $x^l_{i,j}$ é o valor do fragmento da entrada quando centralizada na posição (i,j) da l-ésima camada, já os valores de w^l_k e b^l_k são o vetor de pesos e bias, respectivamente (LI, F.-F.; KARPATHY; JOHNSON, 2015).

$$z^l_{i,j,k} = (w^l_k)^T x^l_{i,j} + b^l_k \tag{1}$$

Depois deste cálculo, ainda é necessário incluir a aplicação da função de ativação não-linear, que pode ser chamada de $f()$, então se apresenta a equação 2 em que se aplica a função de ativação, assim, obtém-se o valor final da saída da camada de convolução, $a^l_{i,j,k}$.

$$a^l_{i,j,k} = f(z^l_{i,j,k}) \tag{2}$$

O objetivo principal da camada de *pooling* é reduzir o número de parâmetros do tensor de entrada. Isto é feito por meio da extração de características representativas do tensor de entrada, assim, reduzindo o custo computacional do processo tornando o modelo mais eficiente, pois o produto desse processo é um tensor menor que o de entrada. Existem dois tipos de *pooling*, o *max pooling* e o *average pooling* (BOUREAU; PONCE; LECUN, 2010; WANG, T. *et al.*, 2012).

O *max pooling* é computado por um kernel de tamanho *n*n* que é movido através do tensor e para cada posição o valor máximo é tomado, desta forma, construindo um novo tensor de tamanho reduzido, porém, ainda com as características principais do tensor de entrada, como pode ser visualizado na Figura 6.

Já o *average pooling* é calculado com mesmo o kernel de tamanho *n*n* que é movido através do tensor, mas agora, ao invés de se tomar o maior valor que o kernel contem, o valor médio é tirado. Então o *pooling* reduz a resolução da imagem em altura e largura, mas o número de canais (profundidade) continua o mesmo. Matematicamente, o processo de *pooling* pode ser representado pela equação 3, em que $a^l_{.,.,k}$ é o mapa de características e $\mathbb{R}_{i,j}$ é uma vizinhança local, centrada na posição (i,j).

$$y^l_{i,j,k} = pool(a^l_{m,n,k}) \forall (m,n) \in \mathbb{R}_{i,j} \tag{3}$$

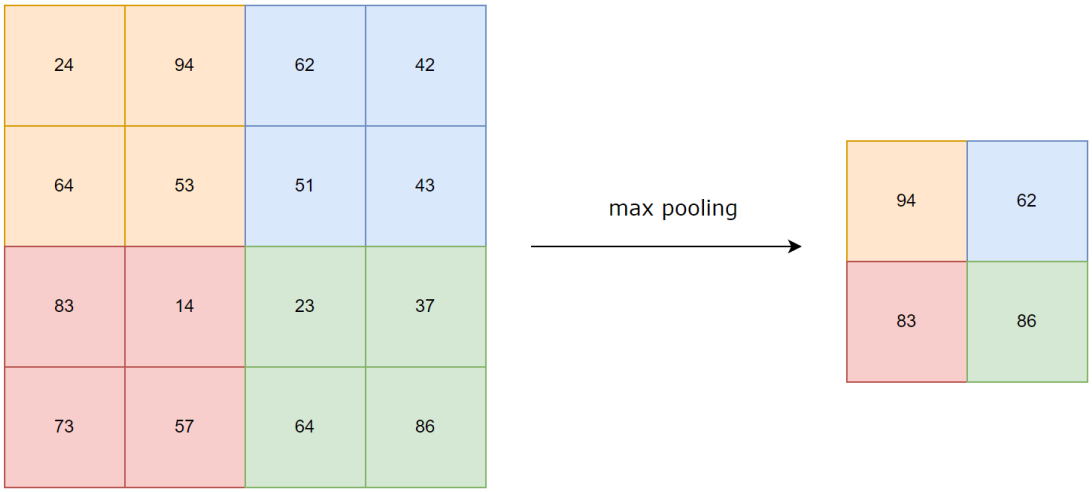


Figura 6 – Exemplo de iteração de max pooling.

A camada *fully connected* é a camada que contém as ligações entre todos os neurônios de duas determinadas camadas, porém, não há conexão entre neurônios de uma mesma camada. Sendo assim, a camada *fully connected* pode ser interpretada como uma MLP.

A entrada desta camada advém dos processos de convolução e *pooling*, portanto, precisa ser achatada, isto é, o tensor de entrada é transformado em vetor. Posteriormente, ocorre o cálculo conforme discorrido na seção 2.2.

As funções de ativação, ou de transferência, podem ser empregadas com diferentes propósitos, os principais são aplicar não linearidades e formatar os dados para saída da CNN.

As não linearidades são aplicadas a fim de estabelecer um ambiente mais favorável para a exploração de características profundas (CHOLLET, 2021). Caso não haja não linearidades entre as camadas de uma rede, o teorema da generalização universal não se aplica, pois o poder de aproximação do modelo é degradado tal qual ao de um com apenas uma camada (HORNIK; STINCHCOMBE; WHITE, 1989).

Já as funções de ativação aplicadas na camada de saída da CNN tem como objetivo adequar os dados à um determinado propósito. Por exemplo, uma rede de classificação pode utilizar a *Softmax* para transformar as saídas em probabilidades, ou seja, diz quão certo o modelo está que determinada imagem pode ser classificada em determinada classe.

A função de transferência comumente utilizada para construir não linearidades

é a Unidade Linear Retificada (ReLU), para operação não linear, equação.4. Seu comportamento toma sentido de retornar 0 para valores menores ou iguais a zero, e o valor original caso seja maior que zero.

$$ReLU(x) = max(0, x) \tag{4}$$

A função logística é usada para a classificação binária e a Softmax é usada para classificação de múltiplos rótulos, e são usados para obter as probabilidades de a entrada estar em classe particular, ou seja, fazer realmente a classificação do objeto.

Uma vez que a imagem passou por todas as etapas da CNN e se gerou um valor de saída, a função de perda é calculada a fim de medir o desempenho do modelo e ditar o processo de treinamento.

Diferentes problemas exigem diferentes funções de perda, porém, a maioria é baseada na **cross-entropy loss function**, equação 5 em que M é o número de classes, $\hat{y}_c$ é a previsão do modelo para a classe c, e y_c é o valor verdadeiro. A família derivada dessa função tem como objetivo quantificar a entropia entre duas distribuições.

$$-\sum_{c=1}^{M}(y_c \cdot \log(\hat{y}_c)) \tag{5}$$

Um problema binário exige uma função de perda binária, como a *binary cross-entropy*, e um problema com múltiplos rótulos pode ser enfrentado com a *Categorical Cross-Entropy* ou a *Sparse Categorical Cross-Entropy*, dependendo da forma com que os rótulos estão organizados.

Para que uma rede neural tenha melhores resultados no processo de classificação, é importante que ela esteja sempre atualizada e que o desempenho aumente, então o uso de otimizadores é essencial, comparando o resultado obtido e o esperado. O Objetivo principal da otimização é reduzir a zero o valor do *cross-entropy*, que é sempre positivo e só se torna zero quando o resultado obtido é o mesmo do esperado. O algoritmo de otimização mais conhecido é o **Gradiente Descendente** onde é usado os parâmetros θ e $\mathbb{R}^d$ afim de minimizar a função objetivo $J(\theta)$. Os parâmetros são atualizados na direção contrária da função objetivo e a taxa de aprendizado η é usada para determinar o tamanho dos passos afim de atingir o mínimo global. O principal problema do uso desse otimizador é o tempo necessário para atualização de pesos e viéses.

Para resolver o problema do tempo, outro método de otimização pode ser usado, chamado *Stochastic Gradient Descent* (SGD) (SUTSKEVER *et al.*, 2013), onde a atualização é feita para cada amostra $x^{(i)}$ e cada rótulo $y^{(i)}$, como pode ser visto na equação 6, uma a cada vez, o que torna-o mais rápido do que o Gradiente Descendente, porém como são diversas atualizações a variância é alta e os resultados sofrem muitas flutuações. O otimizador SGD Momentum (RUDER, 2016) adiciona uma fração γ no vetor de atualização de pesos da Eq 6, fazendo isso acelera o método SGD e auxilia na redução das oscilações do algoritmo.

$$SGD = \theta - \eta \cdot \nabla_\theta \cdot J(\theta; x^{(i)}, y^{(i)}) \tag{6}$$

Por fim, outro algoritmo de otimização que é importante citar é o ADAM (KINGMA; BA, 2014), ele calcula a taxa de aprendizado adaptativa para cada parâmetro da rede e mantém a média exponencial dos gradientes anteriores (TAQI *et al.*, 2018). Nas Equações 7 e 8 temos m_t e v_t são as estimativas do primeiro momento, a média, e o segundo momento, a variação não centralizada dos gradientes e β_1 e β_2 representam as taxas de decaimento.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{7}$$

$$v_t = \beta_2 m_{t-1} + (1 - \beta_2) g_t \tag{8}$$

Mesmo com o ajuste de pesos durante a retropropagação, afim de gerar melhores resultados no modelo, alguns problemas podem surgir no treinamento de uma rede neural, uma delas é o *overfitting*, ou em português sobreajuste. O sobreajuste é uma característica que ocorre quando o modelo aprende muito bem com uma base de dados e atinge ótimos resultados mas não consegue generalizar para novas entradas, é utilizando mais imagens, ou seja, quanto maior o dataset utilizado, melhor o algoritmo pois muito provavelmente seu resultado não estará acostumado com um pequeno grupo de imagens, o que gera o sobreajuste. Quando não existe um banco de dados extenso, é necessária a utilização de métodos de **data augmentation** (LI, W. *et al.*, 2018), que faz alguns processos com as imagens já existentes no dataset e gera outras novas imagens de treino e teste.

Outra maneira de sobrepujar o problema de sobreajuste é aplicar a técnica

de *transfer-learning*, transferência de aprendizado, em que o modelo utilizado já foi previamente treinado em um conjunto de dados extenso, sendo necessário apenas treinar a rede de classificação, e se necessário, realizar ajustes finos nas camadas mais profundas da rede de extração de características (TORREY; SHAVLIK, 2010; WEISS; KHOSHGOFTAAR; WANG, D., 2016).

É comum encontrar aplicações de CNNs com grandes bancos de dados, entrando na área de *Big Data*, isso favorece o quanto o modelo vai aprender, porém trás um custo computacional elevado, sendo muitas vezes necessário o uso de computadores mais potentes. Além do fator de quantidade total de imagens no dataset, outra coisa que deve ser considerada é o balanceamento entre as classes, ou seja, um banco de dados que contém 70% de uma classe A e 30% de uma classe B necessita de algumas etapas de tratamento, como o próprio data augmentation na classe B.

Classificação de imagens médicas para diagnóstico de doenças (YADAV; JADHAV, 2019), classificação de imagens de documentos (KANG *et al.*, 2014), ou segmentação de raio-x (BULLOCK; CUESTA-LÁZARO; QUERA-BOFARULL, 2019) são algumas das aplicações de redes neurais convolucionais.

O ImageNet (RUSSAKOVSKY *et al.*, 2015) é um dataset com mais de 1.000 categorias de imagens usado para pesquisas e competições de reconhecimento de objetos e as mais de 14 milhões de imagens foram anotadas à mão. Desde 2010 o projeto ImageNet faz uma competição anual chamada *ImageNet Large Scale Visual Recognition Challenge*, mas o desafio de 2012 ganhou destaque na área de *deep learning*. Uma rede neural convolucional chamada **Alexnet** (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) que alcançou um erro de 15,3%, mais de 10,8% a menos que o segundo colocado da competição naquele ano. A AlexNet continha oito camadas, as cinco primeiras camadas eram convolucionais, usando de *max pooling*, e as três últimas eram camadas *fully connected*, mas essa taxa de erro também foi possível pelo uso de unidades de processamento gráfico (GPU) durante o treinamento, algo essencial quando trata-se de *deep learning*.

2.2.1.1 DETECÇÃO DE OBJETOS

Tratando de CNNs e imagens, outro ponto importante a ser abordado é a tarefa de **detecção de objeto**, que consiste na previsão da localização do objeto na imagem,

juntamente com a classificação da área prevista.

Para que o algoritmo aprenda a localizar o objeto, é feito a fase de anotação, que serve para colocar retângulos ao redor do objeto de interesse, esses retângulos são chamados de caixas delimitadoras.

A fase de anotação é pertinente para adequar conjuntos de dados para a filosofia da detecção de objeto, uma vez que o processo de aprendizado é supervisionado. O que foi feito no trabalho (KOITKA *et al.*, 2018), em que os autores utilizam técnica de anotação identificar e localizar áreas de ossificação em imagens médicas.

Depois de delimitadas as caixas ao redor do objeto de interesse, as informações ao redor do objeto não precisam estar presentes, afim de melhorar os resultados do modelo, e para isto temos o *Region of Interest* (RoI). Então a ideia principal por trás do RoI é descrita no estudo de (BOUREAU; PONCE; LECUN, 2010), que demonstra por meio de camadas de *pooling*, que algumas regiões nas imagens são preservadas enquanto detalhes irrelevantes são removidos, mantendo único mapa de características dos objetos de propostas na imagem.

O *pooling* de RoI foi proposto por (GIRSHICK, 2015), onde o autor usa o *pooling* máximo para converter recursos dentro de um RoI em um mapa de características, fornecendo um algoritmo de estágio duplo de localização do objeto candidato que aprende a classificar e refinar as localizações espaciais de objetos na imagem.

Com o advento do arquitetura Fast Region-based Convolutional Network (Fast R-CNN), método (GIRSHICK, 2015), outras abordagens usando a mesma estratégia de RoI para segmentar objetos em imagens médicas podem ser observadas em (WANG, S.; YANG, D. M. *et al.*, 2019). Os autores fornecem uma instrução rápida para usar o aprendizado profundo em análise de imagens de patologia, melhorando o desempenho de segmentação.

É importante notar que os objetos RoI não têm tamanho idêntico, o que significa que vários objetos têm formas, tamanhos e localizações diferentes no mapa de características. Os RoI são submetidos a um processo de quantização que transforma a grande entrada em um vetor discreto onde a camada totalmente conectada classificará este vetor (HE *et al.*, 2017).

Os detectores de dois estágios são caracterizados pela separação do processo de geração da proposta da localização dos objetos na imagem, do processo de classificação das propostas. A arquitetura R-CNN conta com a *Region Proposal Network*

para gerar as propostas de caixas limitadoras, que em seguida, são classificadas pela rede de classificação. Já os detectores de um estágio não possuem a separação entre geração de proposta e classificação. Ao fim do processo de extração de características, cada mapa é uma proposta, que segue para ser processado por duas subredes: classificação e de regressão; ao passo que a proposta é gerada ao mesmo tempo em que é classificada (LIU, W. *et al.*, 2016).

R-CNN provem de Region-Based Convolutinal Neural Networks, ou em português Redes Neurais Convolucionais baseadas em Região, e essa rede tem a proposta de *Seletive Search*, que reduz o número de caixas delimitadoras que são fornecidas para o classificador, fornecendo cerca de 2000 propostas de região (GIRSHICK *et al.*, 2014), depois essas 2000 propostas de regiões são distorcidas em um quadrado e viram a entrada de uma CNN, que produz um vetor de 4096 dimensões de saída.

A CNN atua como extratora e a camada de saída consiste nos recursos extraídos da imagem, que são a entrada da SVM (*Support Vector Machine*), que classifica a presença do objeto dentro daquela região candidata. Além de prever a presença do objeto dentro da área candidata, esse algoritmo também prevê quatro valores de deslocamento para aumentar a precisão da caixa delimitadora.

A R-CNN trás melhorias em sua arquitetura para que exista uma concentração de uma região por vez na imagem, porém ainda tem problemas, como o tempo de treinamento da rede total, já que classifica 2000 propostas de região por imagem, levando cerca de 47 segundos por imagem, então não pode ser implementada em tempo real, além de que o algoritmo de busca seletiva é fixo, ou seja, nenhum aprendizado ocorre.

Do mesmo autor da R-CNN, a **Fast R-CNN** (GIRSHICK, 2015) resolve um dos principais problemas da R-CNN, o tempo de detecção da imagem, por isso é chamada de R-CNN rápida. A abordagem dela é semelhante à anterior, porém ao invés de alimentar as propostas de região para a CNN, alimentamos a imagem de entrada para a CNN para gerar um mapa de características convolucionais. A partir do mapa de recursos convolucionais, é possível identificar a região proposta e transformá-la em quadrados para depois, usando uma camada de pooling, remodelar em um tamanho fixo que possa ser a entrada de uma camada *fully conected*. A partir do vetor de recursos RoI, é usado uma função Softmax para prever a classe da região proposta e também os valores de deslocamento para a caixa delimitadora. A principal razão para que a

Fast R-CNN seja mais rápida que a R-CNN é porque ela não precisa utilizar as 2000 propostas de região todas as vezes, ao invés disso a convolução é feita apenas uma vez por imagem e o mapa de características é gerado a partir dela.

Em 2015, Shaoqing et al (REN *et al.*, 2015) criou um algoritmo semelhante ao Fast R-CNN, nomeado de Faster R-CNN, que faz a detecção de objetos eliminando a busca seletiva do algoritmo e faz com que a rede aprenda as propostas da região. Neste modelo, a imagem é fornecida como entrada para a rede convolucional, que gera um mapa de recursos convolucionais, e ao invés de utilizar a busca seletiva, as propostas da região são identificadas e uma rede separada é utilizada para prever estas propostas. Depois as propostas da região prevista são remodeladas usando uma camada de pooling e RoI, usadas para classificar a imagem dentro da região proposta e prever os valores de deslocamento das caixas delimitadoras. A rede Faster R-CNN em um Test-Time Speed teve ótimos resultados comparados aos das suas irmãs R-CNN e Fast R-CNN, fazendo a detecção de um objeto em cerca de 0.2 segundos, ou seja, pode ser usada para detecção de objetos em tempo real.

Single Shot Detector (SSD) ou detector de disparo único, é um tipo de detector onde tira apenas uma foto para detectar vários objetos presentes em uma imagem usando as caixas múltiplas. As camadas de convolução desse modelo avaliam caixas de diferentes relações de aspectos em cada local, em vários mapas de recursos com diferentes escalas. As caixas múltiplas somam cerca de 8732 caixas, o que ajuda a encontrar a caixa padrão que melhor se encaixa na caixa delimitadora real, que contém o objeto (LIU, W. *et al.*, 2016).

O modelo mais conhecido de detecção de disparo único é o YOLO (*You Only Look Once*) (HUANG; PEDOEEM; CHEN, C., 2018) e ao invés de usar uma região para localizar o objeto na imagem, sem ver a imagem num contexto completo como as redes convolucionais da família R-CNN, utiliza da previsão de diversas caixas delimitadoras pela imagem e calcula a probabilidade de classe para cada caixa. Então a imagem é dividida em uma grade $S \times S$ e a rede produz uma probabilidade de classe e valor de deslocamento para cada quadrado da grade, depois as caixas delimitadoras com a probabilidade são selecionadas e usadas para localizar o objeto da imagem. Mesmo tendo uma visão geral de imagem e ser aparentemente melhor que a R-CNN, 45 quadros por segundo, a YOLO é limitada quando existem pequenos objetos, como diversos pássaros voando juntos.

RetinaNet (LIN; GOYAL *et al.*, 2017) é um modelo de detecção de objetos de um estágio que utiliza a função de perda **Focal Loss** para lidar com o desequilíbrio de classe (*class imbalance*) durante o treinamento. A *Focal Loss* aplica um termo de modulação à perda de *cross-entropy* a fim de concentrar o aprendizado em contra-exemplos específicos. RetinaNet é uma rede unificada, onde existe uma rede de *backbone* e duas sub-redes específicas para tarefas. O *backbone* é responsável por calcular o mapa de características convolucionais de toda a imagem de entrada e é uma rede convolucional autônoma. A primeira sub-rede realiza a classe convolucional do objeto na saída do *backbone*; a segunda sub-rede realiza a regressão da caixa delimitadora de convolução. Essas duas sub-redes adotam um design simples proposto pelo autor para detecção densa de nível único.

Explicando em melhores termos a *Focal Loss*, essa função é uma *cross-entropy* dinamicamente escalada. À medida que o coeficiente de confiança aumenta, o fator de escala modula a perda para zero. O fator de escala ameniza o impacto das amostras com alto fator de confiança alto, assim, focando nos exemplos mais difíceis. Sendo assim, a *Focal Loss* modula para baixo a influência dos exemplos fáceis - segundo plano - na função de perda, o que força o processo de aprendizagem a focar nos exemplos difíceis, ou seja, nos objetos de interesse.

Voltando ao *backbone* da RetinaNet, ela utiliza um modelo de ResNet-50 e mais as suas outras sub-redes. A FPN, *Feature Pyramid Network*, é similar a camadas convolucionais estacadas, em que cada camada extrai características em diferentes escalas e níveis semânticos, um conceito importante na tarefa de reconhecimento de objetos. Em uma visão geral, a ResNet-50 é responsável por extrair as características da imagem da forma convencional, caminho debaixo para cima, e em paralelo é construído um caminho decima para baixo, ambos caminhos são conectados por ligações residuais.

Juntamente com a FPN, duas sub-redes são utilizadas. Uma para predizer a localização do pulmão se desenhando uma caixa limitadora, e outra pra classificar o pulmão dentro desta caixa. A Figura 7 ilustra a arquitetura discutida.

2.2.2 MÉTRICAS PARA AVALIAÇÃO DE MODELOS

Quando se constrói modelos de *machine learning* é necessário avaliar seu desempenho em diferentes fases de desenvolvimento, principalmente na área da saúde,

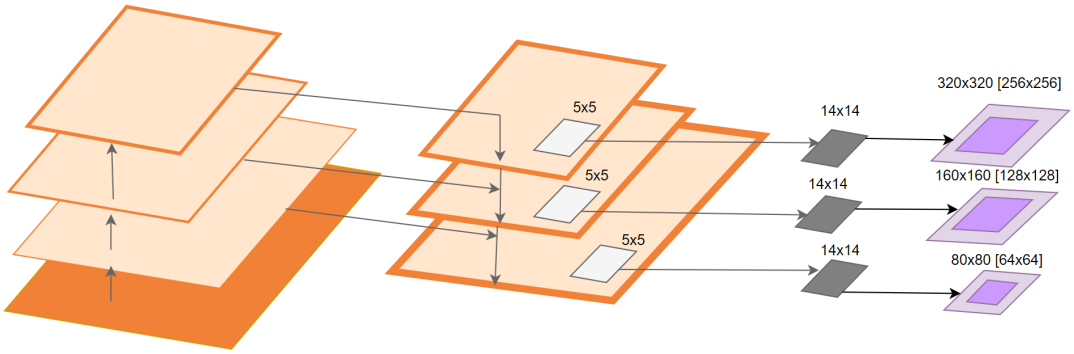


Figura 7 – FPN para segmentação e reconhecimento de objetos.

onde vidas são diretamente envolvidas e dependem desses resultados. Porém, cada tipo de projeto precisa utilizar a métrica que seja mais apropriada, e caso sejam mal escolhidas é impossível dizer se o modelo está realmente atendendo os requisitos ou está enviesado.

Em modelos de classificação, surgem quatro entidades: falsos positivo (FP), falso negativo (FN), verdadeiro positivo (TP) e verdadeiro negativo (TN). E destas entidades, derivam-se outras métricas, como acurácia, precisão, sensitividade, especificidade e a matriz de confusão

É possível enquadrar essas entidade em duas grandes classes, onde a ocorrência ou não de um determinado evento é anotada, denominado de positiva (P) ou negativa (N), quando existe a classificação correta denomina-se de verdadeiro ou *true* (T), e quando o modelo erra a classificação é chamado de falso ou *false* (F). Por isso temos os quatro grupos, onde:

- ***False Positive* (FP)**: Indica quantos foram classificados como positivo da maneira errada, ou seja, o modelo sugeriu que seria positivo e na verdade era negativo.
- ***False Negative* (FN)**: Indica quantos foram classificados como negativos da forma errada, então o modelo previu que seria negativo mas na realidade era positivo.
- ***True Positive* (TP)**: Indica quantos foram classificados como positivo de maneira correta, ou seja, o modelo achou que era positivo e realmente era.
- ***True Negative* (TN)**: Indica quantos foram classificados como negativos corretamente, ou seja, o modelo classificou como negativo e era negativo.

A matriz de confusão é provavelmente uma das métricas mais comuns de

avaliação de modelos de classificação em *machine learning* (SAMMUT; WEBB, 2011). A morfologia da matriz consiste em evidenciar os valores de falsos positivos, falsos negativos, verdadeiros positivos e verdadeiros negativos, com isso é possível avaliar o desempenho da rede, em questão de classificação de itens corretos e incorretos em cada classe, o que facilita o a construção de uma visão ampla da performance da proposta.

A figura 8 ilustra um exemplo de matriz de confusão, onde se tem os valores preditos e os valores reais. Então quando um valor predito é "sim"e o valor real é "sim", gera um **TP**, já a direita temos um valor predito como "não"mas o real era "sim", então ele é um **FN**. Na próxima linha, o valor real é "não"e existe a variação de "sim"no primeiro quadrado no valor predito, gerando um **FP**, e na direita quando o valor predito é "não", gera um **TN**.

A matriz de confusão pode exibir os valores absolutos dos acertos e erro para cada classe, sendo a soma total a quantidade de indivíduos preditos pelo modelo, ou em termos de proporção, em que os valores absolutos são divididos pelo valor total de indivíduos de cada classe respectiva.

A acurácia é a métrica responsável por dizer quantos exames foram realmente classificados de forma correta, independente da sua classe, ou seja, quantifica o poder de generalização total do modelo. A equação 9 expressa o calculo da acurácia, em que é representada pela razão entre o que o modelo realmente acertou e a soma total dos indivíduos.

$$Acurcia = \frac{TP + TN}{TP + FP + TN + FN} \tag{9}$$

Sensibilidade ou em inglês *recall*, é a métrica que avalia a capacidade do modelo em classificar os exemplos que são originalmente positivos tendo em vista todos o conjunto de elementos originalmente positivos. O método de obtenção está presente na equação 10.

$$Sensibilidade = \frac{TP}{TP + FN} \tag{10}$$

A especificidade mede a capacidade do modelo em classificar os exemplos que são originalmente negativos tendo em vista todos o conjunto de elementos originalmente negativos. A equação 11 mostra seu cálculo.

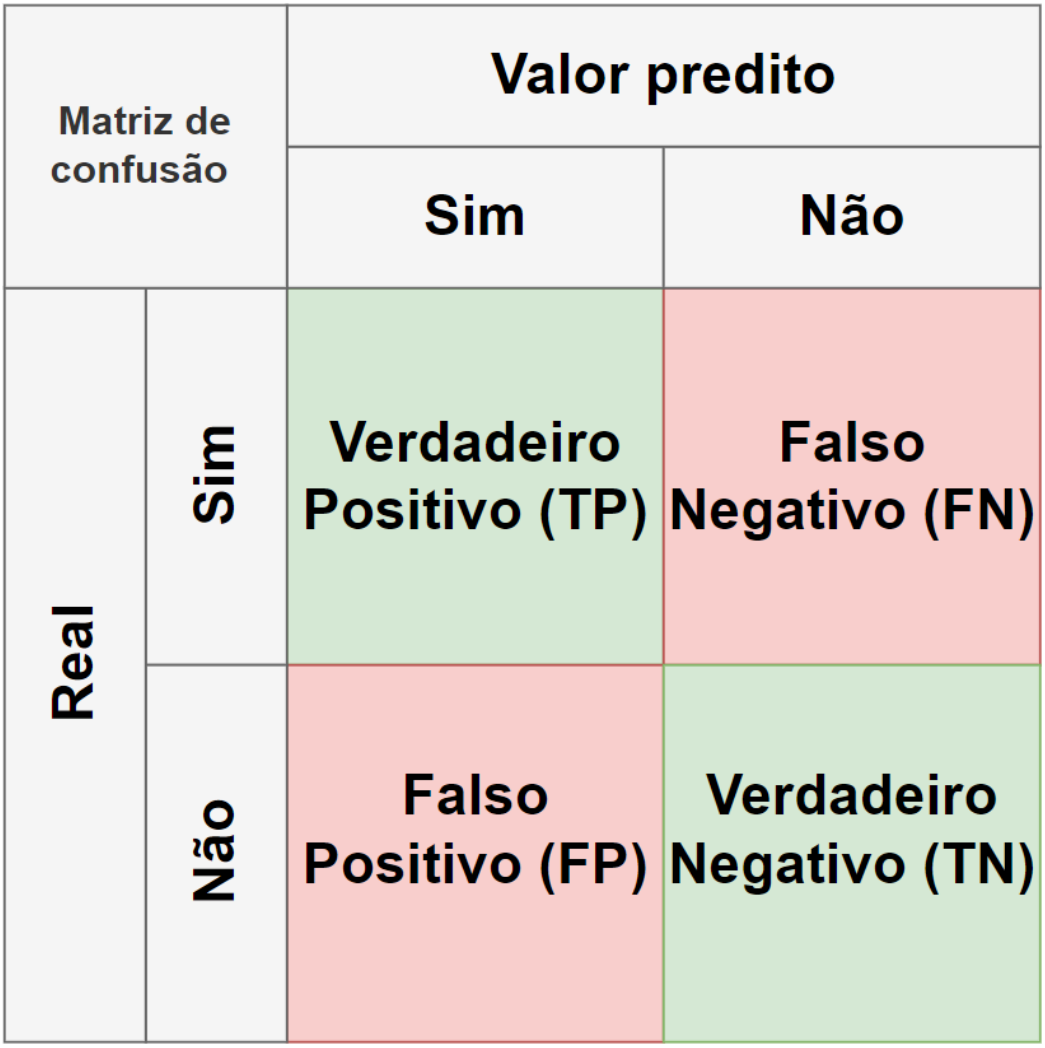


Figura 8 – Matriz de confusão.

$$Especificidade = \frac{TN}{FP + TN} \tag{11}$$

A precisão é definida pela razão entre a quantidade de exemplos classificados corretamente como positivos e o total de positivos, tanto verdadeiros quanto falsos, como pode ser visto na equação 12. Esta métrica mede a capacidade do modelo em em predizer corretamente os casos positivos.

$$Preciso = \frac{TP}{TP + FP} \tag{12}$$

Do inglês, *Intersection over Union* (IoU), expressa a sobreposição entre a caixa delimitadora prevista pelo modelo e a caixa delimitadora real (*Ground Truth*), como ilustrado na Figura 9.

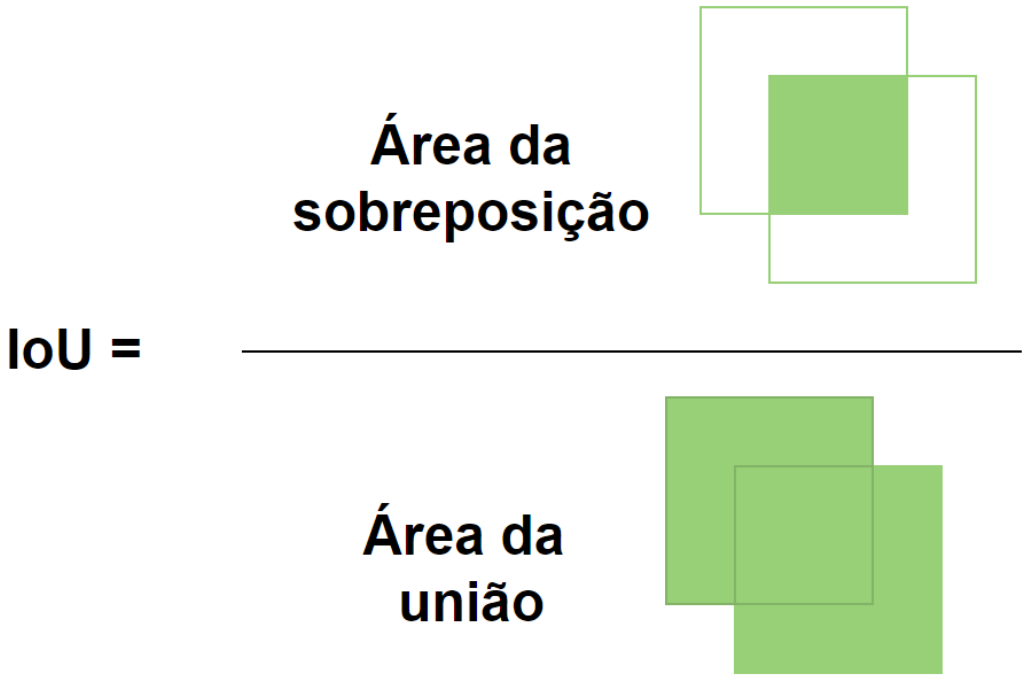


Figura 9 – Representação de como é feito a IoU.

Esta métrica influencia outras, como a Precisão Média, no sentido de estabelecer um escore de confiança de conter um objeto. De forma que se seu valor for menor que um valor limiar para determinada caixa delimitadora, esta caixa é desconsiderada (FP), e se maior (TP), a caixa delimitadora e seu conteúdo são utilizados para o calculo de outras métricas.

Do inglês *Average Precision - AP*, consiste na área sob a curva entre $Preciso \times Sensibilidade$. O valor da sensibilidade aumenta conforme se diminui os valores de falsos negativos, porém, a precisão diminui conforme os falsos positivos aumentam e sobe quando os verdadeiros positivos aumentam, fazendo com que o gráfico tenha um formado de "zigue-zague"

Quando o classificador opera de forma a classificar múltiplas classes, torna-se inviável avaliar a performance do modelo tomando somente o AP de cada classe. Para tanto, a precisão média média (mAP) toma todos os valores de AP e faz a média entre eles.

Como exposto anteriormente, a métrica IoU tem influencia no calculo da AP e

mAP, pois estabelece um limiar as caixas consideradas como verdeiro ou falso positivo, daí surgem as métricas mAP50 e mAP75, em que se arbitra um valor limiar através do IoU de 0.5 e 0.75, respectivamente.

2.3 TRABALHOS RELACIONADOS

Em (KESIM; DOKUR; OLMEZ, 2019) os autores propõem um modelo de CNN de pequeno porte para a classificação de imagens CXR em vez da arquitetura clássica de grande porte. Isso proporciona um caminho rápido e exige um baixo custo computacional. As imagens CXR são atribuídas a doze classes com aproximadamente 86% de taxa de sucesso em um curto período de tempo devido à pequena estrutura de rede. Uma boa atuação para dar suporte aos médicos.

Os autores em (LI, X. *et al.*, 2020) implementaram um modelo CNN usando abordagem de *transfer learning* para detecção de tuberculose usando imagens CXR fornecendo uma precisão de 85,68%. Em (CHOUHAN *et al.*, 2020), os autores aplicam cinco modelos em *deep-transfer-learning-based* como um conjunto para detectar pneumonia em imagens CXR. Os resultados obtidos mostram uma precisão de 96,4%. Outra abordagem interessante é desenvolvida por (BHANDARY *et al.*, 2020), onde os autores modificam o modelo AlexNet para detectar pneumonia através de imagens CXR, propondo um novo filtro de *threshold* e uma estratégia de conjunto de características resultando em uma precisão de classificação de 96%.

Alguns dos métodos de ML foram desenvolvidos especialmente para auxiliar os profissionais de saúde no diagnóstico de imagens médicas COVID-19 (ZHANG *et al.*, 2020). Recentemente, (ISMAEL; ŞENGÜR, 2021) aplica três métodos CNN profundos para detectar COVID-19 com base em imagens CXR: duas abordagens de aprendizagem de transferência para extração de recursos e ajuste fino; e um modelo CNN treinado ponta a ponta. Support Vector Machines (SVM) são usados para classificar recursos profundos, juntamente com diferentes funções do kernel. Os modelos CNN profundos pré-treinados também são usados para o ajuste fino, e oito descritores locais bem conhecidos são endereçados. Os autores abordam um conjunto de dados com 180 COVID-19 e 200 imagens CXR saudáveis. Outra abordagem para imagens CXR são propostas por (KHUZANI; HEIDARI; SHARIATI, 2021). Os autores usam um método de redução de dimensionalidade para criar um conjunto de recursos ótimos de imagens

CXR para classificar os casos COVID-19 e não COVID-19. Além disso, um pequeno conjunto de dados de imagens CXR é considerado o que resulta em uma precisão desejável.

Algumas abordagens interessantes também são desenvolvidas considerando a TC. O trabalho apresentado em (WANG, S.; ZHA *et al.*, 2020) aplica um modelo de *deep learning* pré-treinado chamado DenseNet 121 para imagens CT para detectar COVID-19 com 81,24% de precisão. Além disso, o trabalho (ZHANG *et al.*, 2020) propõe a segmentação da lesão pulmonar em imagens de TC usando um modelo classificador ResNet-18 considerando três classes: COVID-19, pneumonia e normal. Seus resultados apresentam uma precisão de 92,49%. Para imagens LUS, os autores em (ROY *et al.*, 2020) abordam um pequeno conjunto de dados para analisar infecções por COVID-19 (11 pacientes) aplicando modelos de aprendizagem profunda.

Tabela 1 – Lista de trabalhos relacionados.

Autor	Uso e tipo de imagem	Tipo de rede	Precisão
KESIM; DOKUR; OLMEZ,2019	Classificação de pulmões em CXR	CNN pequeno porte	86%
LI, X. et al., 2020	Detecção de tuberculose em CXR	CNN com *transfer learning*	85.68%
CHOUHAN et al., 2020	Detecção de pneumonia em CXR	*deep-transfer-learning-based*	96.4%
BHANDARY et al., 2020	Detecção de pneumonia em CXR	AlexNet modificada, com novo *threshold*	96%
ISMAEL; ŞENGÜR, 2021	Detecção de COVID-19 em CXR	Três métodos de CNN	-%
KHUZANI; HEIDARI; SHARIATI, 2021	Detecção de COVID-19 em CXR	Métodos de redução de dimensionalidade	-%
WANG, S.; ZHA et al., 2020	Detecção de COVID-19 em TC	DenseNet 121	81.24%
ZHANG et al., 2020	Segmentação de lesão pulmonar em TC	ResNet-18	92.49%
ROY et al., 2020	Infecções por COVID-19 em LUS	modelos de aprendizagem profunda	96.4%
Este trabalho, 2021	Detecção de COVID-19 em CXR	RetinaNet (ResNet-50 + 2 sub-redes)	-%

Fonte: autoria própria (2021).

3 MATERIAIS E MÉTODOS

Esta seção discute o fluxo de trabalho proposto para detecção e classificação de objeto através do modelo de aprendizado profundo *RetinaNet*.

A Figura 10 ilustra o fluxo de processamento da abordagem proposta. Primeiramente, imagens das classes COVID-19 e Normal são manualmente selecionadas. Posteriormente, as regiões de interesse são anotadas nas imagens selecionadas, desenhando-se uma caixa delimitadora e se atribuindo a respectiva classe. Com as imagens e suas anotações o modelo é treinado e posteriormente avaliado em diferentes conjuntos de dados. A avaliação da performance mede a capacidade da rede de detectar e classificar pulmões dentre as classes propostas.

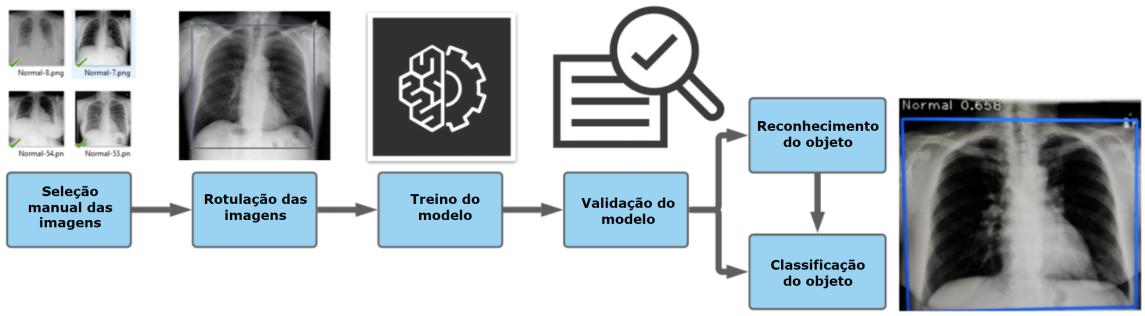


Figura 10 – Fluxo de Trabalho.

3.1 CONJUNTO DE DADOS

Este trabalho utiliza três conjuntos de dados para as diferentes fases de trabalho do modelo. Todos são compostos de imagens de raio-X de tórax no formato PNG, com 299 pixels de altura e largura, e apenas um canal de cor, o cinza.

O primeiro conjunto, é o *COVID-19 Radiography database* [1], em que é composto de quatro classes: Pneumonia Viral, Opacidade Pulmonar, Normal e COVID-19. Os laudos de radiologia utilizados para montar este conjunto de dados foram coletados do *Medical Imaging Databank* na *Valencian Region Medical Image Bank* (CHOWDHURY *et al.*, 2020; RAHMAN *et al.*, 2021). Apenas duas classes destas quatro foram utilizadas, Normal e COVID-19, visto que o trabalho está nas fase inicial.

[1] https://www.kaggle.com/tawsifurrahman/covid19-radiography-database.

O segundo, é um conjunto de dados privado e foi fornecido pelo *HmHospitales* (HM. . . , 2020). Este conjunto contem casos positivos e pendentes para COVID-19, apenas os casos positivos foram utilizados. Este banco de imagens foi disponibilizado através de requisição formal para a rede de espanhola Hm Hospitales.

O terceiro, é um conjunto publico e advém do *National Institute of Health* [2]. As imagens são classificadas em quatorze tipos de achados radiológicos. A metodologia empregada na estruturação das classes foi *Natural Language Programing*, em que as informações foram retiradas de laudos médicos (WANG, X. *et al.*, 2017). Este banco de imagens foi criado antes do ano de 2019, portanto, todas as imagens podem ser classificadas na classe Normal.

Com estes três conjuntos de dados, as imagens de treino, validação e teste são selecionadas. O conjunto de treino e validação é composta por imagens presentes no *COVID-19 Radiography database*. Já o conjunto de testes pelos outros dois, a classe COVID-19 pelo HmHospitales e a classe Normal pelo National Institute of Health.

Dado que o modelo de aprendizado profundo proposto detecta e classifica pulmões em imagens de raio-X de tórax, algumas etapas de preprocessamento se fazem necessárias, como seleção de imagens e o processo de anotação das caixas delimitadoras.

3.2 SELEÇÃO MANUAL DE IMAGEM

O primeiro passo da metodologia proposta é a seleção manual das imagens, em que baseou-se na experiencia de profissionais da medicina [3] para extrair um subconjunto representativo dos bancos de dados. O descarte foi julgado com base na qualidade da imagem de raio-X, na presença de aparatos médicos como eletrodos e sondas, e na presença de texto na imagem. Este processo resultou em 1.500 imagens da classe Normal e 1.500 da classe COVID-19 advindas do banco *COVID-19 Radiography database* para treino e validação, e mais 1.000 imagens da classe COVID-19 do banco *HmHospitales* e outras 1000 do NIH, totalizando 5.000 imagens.

[2] https://nihcc.app.box.com/v/ChestXray-NIHCC.

[3] Os autores são gratos pelo auxilio do profissional da medicina Mayler Olombra.

3.3 ANOTAÇÃO DE IMAGEM

A fase de anotação de imagem é importante para adequar o conjunto de dados para a filosofia proposta: detecção e classificação de objeto. Sendo assim, desenhou-se caixas delimitadoras nos pulmões e se atribuiu a respectiva classe nas 3.000 imagens dos subconjuntos de treino e validação. As imagens do subconjunto de teste não foram anotadas visto que a avaliação na etapa de teste busca apenas medir o poder de classificação do modelo. Esta etapa foi realizada com a ferramenta de código aberto *LabelImg* [4]. A interface do programa pode ser visualizada na figura 11.

O processo de anotação através do *LabelImg* tem como resultado um arquivo XML para cada imagem, em que estes arquivos contêm as coordenadas das caixas limitadoras e sua classe. A Figura 12 ilustra o conteúdo de um arquivo XML.

3.4 DESIGN DO MODELO

O modelo de aprendizado profundo aplicado é o *RetinaNet* (LIN; GOYAL *et al.*, 2017), que é um detector de objetos de um estágio, munido da função de perda chamada *Focal Loss*. A arquitetura *RenitaNet* adota a *Feature Pyramid Network* (FPN) com a ResNet50 como a espinha dorsal do modelo em conjunto com duas subredes,conforme descrito na seção 2.2.1.1.

A característica mais importante da *RetinaNet* é a função de perda, a *Focal Loss*. Esta função busca resolver o problema do desbalanceamento de classes gerados pela presença massiva de segundo plano comparado com os objetos alvo. Como neste trabalho não há a classe de segundo plano, a *Focal Loss* guia o processo de aprendizagem a focar nos casos de difícil detecção de COVID-19, desenhando uma linha mais clara na fronteira de decisão.

A técnica de *transfer-learning* é aplicada para minimizar o custo computacional para treinar o modelo (PAN; YANG, Q., 2009; TORREY; SHAVLIK, 2010), melhorar os resultados e evitar treinos desnecessários. O modelo utilizado é o *RetinaNet* com a ResNet50 [5] pré-treinado no conjunto de dados *Common Objects in Context* (LIN;

[4] https://github.com/tzutalin/labelImg.
[5] https://github.com/fizyr/keras-retinanet.

MAIRE *et al.*, 2014) na tarefa de detecção de 500 diferentes classes, em que se obteve um mAP de 0,4594.

A sub-rede de regressão possui 2.443.300 parâmetros, a de classificação 2.401.810, e o modelo completo 36.403.702 dos quais 12.842.550 são treináveis. A diferença da quantidade entre os parâmetros treináveis e não treináveis se dá pelo congelamento da ResNet50 durante a fase de treino, isto é, os parâmetros não são ajustáveis, pois isto implicaria no ajuste de todos os parâmetros do modelo, o que não se faz necessário visto que a capacidade de extração já aprendida pela ResNet50 pode ser utilizada (WEISS; KHOSHGOFTAAR; WANG, D., 2016).

A implementação do modelo *RetinaNet* foi realizada com a ferramenta *Keras* [6], uma biblioteca de construção de fluxo computacional por blocos.

A entrada do modelo possui cinco dimensões. A primeira consiste na quantidade de elementos que serão computados pela rede em uma única época de treinamento, os lotes de dados (*batch*), a segunda é a largura da imagem, a terceira a altura, já a quarta consiste no canal de cores da imagem, por fim, a quinta dimensão abriga as coordenadas da caixa delimitadora (tamanho do lote, largura da imagem, altura da imagem, canal de cores da imagem, coordenadas da caixa delimitadora). Quando a rede está no formato de treinamento, a classe de cada caixa delimitadora é agregada na quinta dimensão dos dados de entrada.

3.5 AUMENTO NA QUANTIDADE DE IMAGENS POR MEIO DE ALTERAÇÕES RANDÔMICAS

Visto que o processo de anotação de imagem é muito custoso na questão de tempo, uma alternativa para aumentar a quantidade de imagens se dá pela alteração das características das imagens já anotadas. Desta forma, o sobreajuste dos pesos não ocorre facilmente, além de aumentar o poder de generalização do modelo (SHORTEN; KHOSHGOFTAAR, 2019).

As mudanças das características originais das imagens pode ser realizado por meio de operações randômicas, como rotações, translações, cortes, alterações na escala e espelhar a imagem horizontalmente. Todos esses processos foram aplicados

[6] https://keras.io.

na fase de treinamento.

3.6 *DATA LEAKAGE*

Data Leakage é um termo do inglês que se refere a utilização de imagens medicas de um mesmo exame de um determinado pacientes repetidas vezes no mesmo subconjunto de imagens. Ou, a presença do mesmo paciente em diferentes subconjuntos de imagens por meio de exames diferentes. Ambos os casos podem levar a geração de valores otimistas de métricas de avaliação do modelo, ou seja, imputa viés, uma vez que redes neurais convolucionais são capazes de aprender diversas características das imagens (PAPADIMITRIOU; GARCIA-MOLINA, 2010; KAUFMAN *et al.*, 2012).

Este fenômeno pode ocorrer quando não há documentação que discrimine a origem das imagens, nem utilize um sistema de identificação de pacientes e exames consistente. Sendo, muitas vezes, não proposital, pois a há grande complexidade na organização de dados médicos.

Os autores amenizaram os efeitos do *Date Leakage* ao se trabalhar com conjuntos de dados de diferentes épocas e origens, bem como separando de forma a incluir o mesmo paciente em apenas um subconjunto de dados.

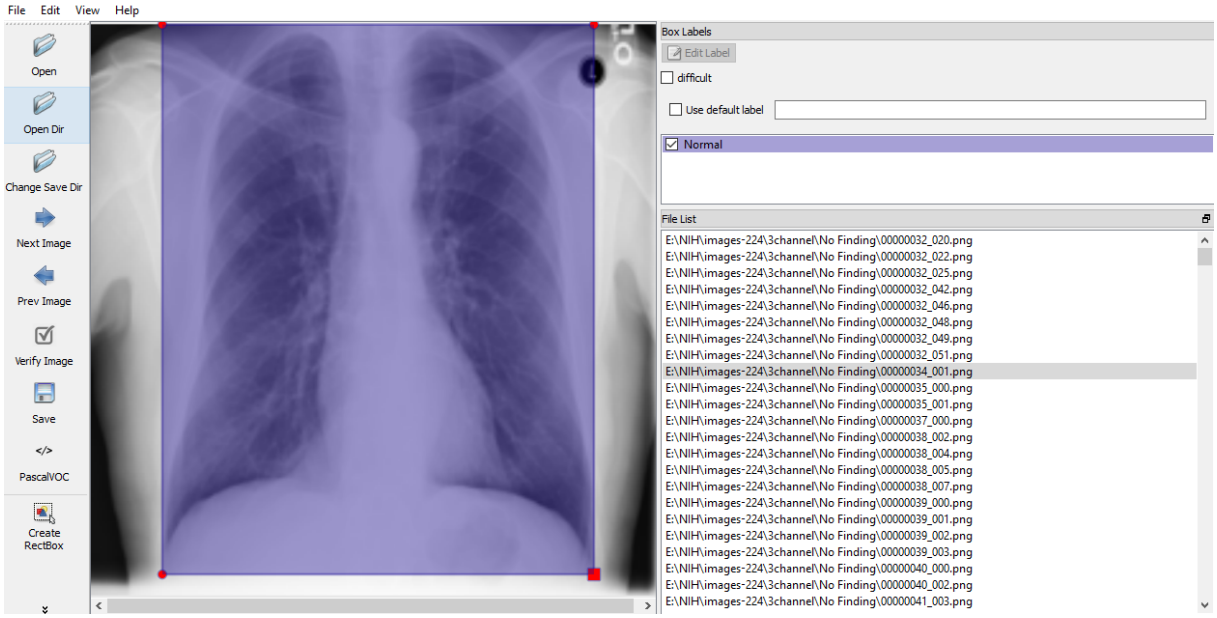


Figura 11 – Exemplo de anotação de imagem com o programa *LabelImg*

File Edit Format View Help

```
<annotation>
        <folder>No Findings</folder>
        <filename>patient00089_study1_view1_frontal.jpg</filename>
        <path>E:\NIH\images-224\3channel\No Finding\00000002_000.jpg</path>
        <source>
                <database>Unknown</database>
        </source>
        <size>
                <width>390</width>
                <height>320</height>
                <depth>3</depth>
        </size>
        <segmented>0</segmented>
        <object>
                <name>NORMAL</name>
                <pose>Unspecified</pose>
                <truncated>0</truncated>
                <difficult>0</difficult>
                <bndbox>
                        <xmin>77</xmin>
                        <ymin>41</ymin>
                        <xmax>304</xmax>
                        <ymax>262</ymax>
                </bndbox>
        </object>
</annotation>
```

Figura 12 – Exemplo da estrutura do conteúdo de um arquivo XML de anotação de imagem.

4 RESULTADOS E DISCUSSÃO

O estudo computacional está organizado em fases de treinamento e teste. A origem dos dados utilizados em cada fase está descrito na seção 3.1. Todas as imagens dos subconjuntos foram selecionadas de forma aleatória e tendo em vista o problema de *Data Leakage*.

4.1 FASE DE TREINAMENTO

A parcela de treinamento é composta de 1.000 casos de COVID-19 e 1.000 casos normais, e a parcela de validação de 500 COVID-19 e 500 Normais. O modelo é treinado por 50 épocas, cada uma com 200 passos. A taxa de aprendizado inicial é de 10^{-6}. O otimizador utilizado é o *Adam*. Cada época de treino gera um novo modelo que é salvo.

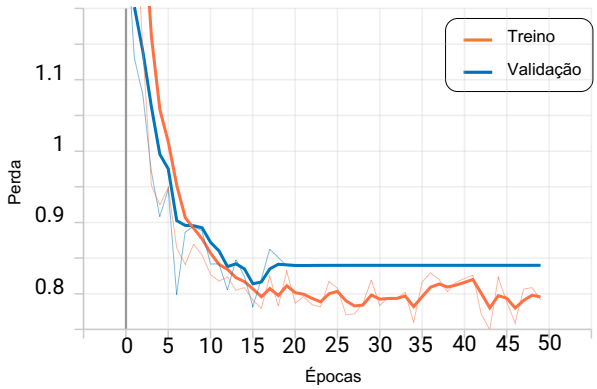
O comportamento do aprendizado do modelo por ser visualizado através dos gráficos de perda. A Figura 13(a) ilustra a perda total do modelo em função das épocas de treinamento. É possível notar que o processo de sobreajuste começa em meados da época 20, pois não há melhora na perda de validação e a perda de treino segue decaindo. O menor valor de perda de validação ocorre na época 15, portanto, esse é o modelo escolhido para gerar as métricas que serão apresentadas posteriormente nesta seção. E, a fim de comparação, o modelo da época 49 também é utilizado.

A perda de classificação, Figura 13(b), e a perda de regressão, Figura 13(c), descrevem o mesmo comportamento da perda total, pois a perda total é a soma da perda de regressão e classificação. A perda de regressão é a *L1 smooth* e a de classificação é a *Focal Loss*.

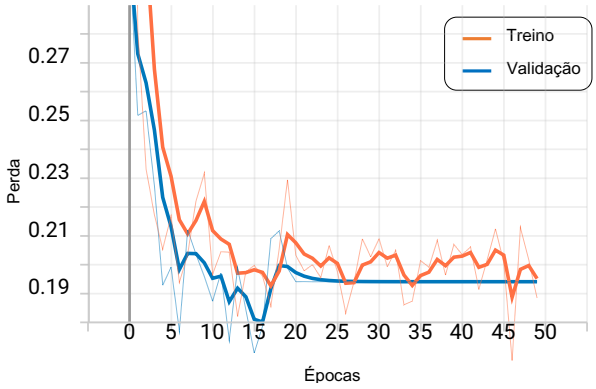
A taxa de aprendizado é variável, como se pode visualizar na Figura 13(d), começando em 10^{-6} e finalizando, na época 49, em 10^{-9}. Esta variabilidade tem como finalidade guiar o passo de aprendizado para o ponto mais baixo da região ótima que está sendo explorada. A intensidade e frequência do decréscimo da taxa de aprendizado é calculada com base na taxa de variação da perda de validação.

Na tabela 2, é possível observar que o modelo 15 é superior ao 49 em todas as métricas. Além disso, ambos modelos atingiram escores menores na classe Normal frente a classe COVID-19.

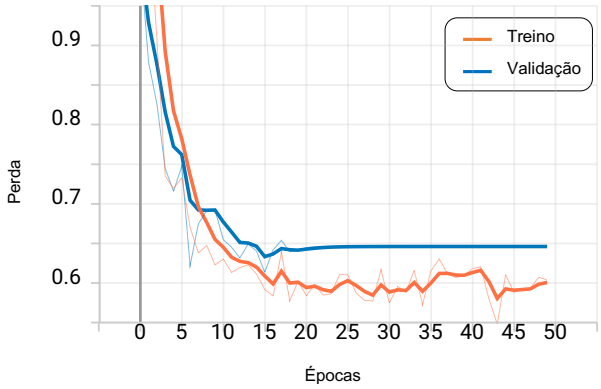
Figura 13 – Gráficos de Perda e Taxa de Aprendizado em Função das Épocas de Treino.



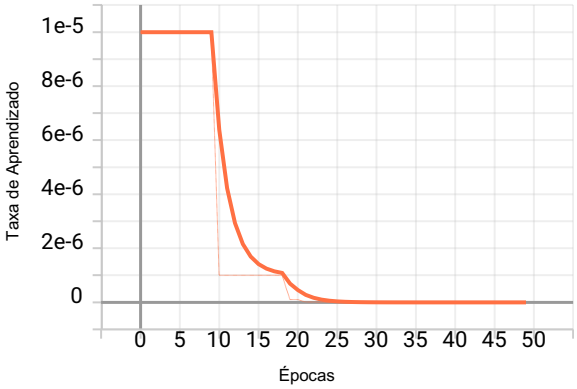
(a) Perda Total.



(b) Perda de Classificação.



(c) Perda de Regressão.



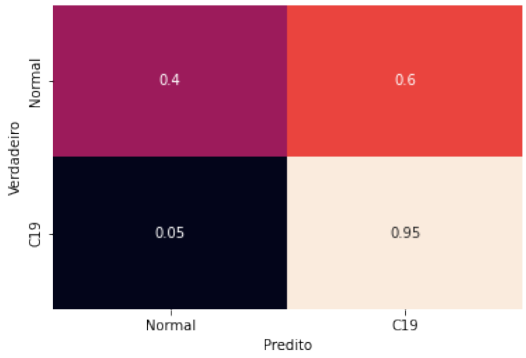
(d) Taxa de Aprendizagem.

Tabela 2 – Comparação de métricas entre os modelos 15 e 49 - I.

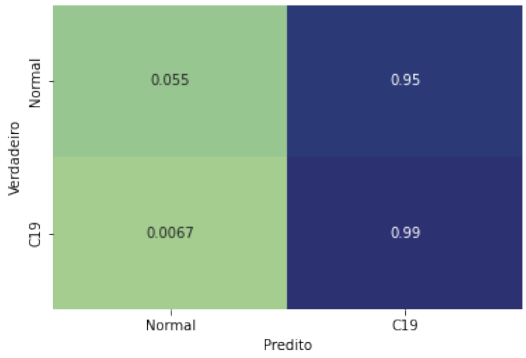
Métrica	Modelo 15	Modelo 49
AP - Normal	0,7848	0,7826
AP - C19	0,8417	0,8239
mAP	0,8133	0,8032

As matrizes de confusão obtidas no subconjunto de validação para os modelos 15 e 49 estão presentes na Figura 14. É nítido que ambos modelos possuem maior facilidade para classificar COVID-19 do que a classe Normal. O modelo 15, figura 14(a), classificou erroneamente 5% dos pulmões com COVID-19 e 60% dos pulmões da classe Normal.

Figura 14 – Matrizes de Confusão de Validação.



(a) Matriz de Confusão do Modelo 15.



(b) Matriz de Confusão do Modelo 50.

Outras métricas podem ser extraídas das matrizes de confusão, como acurácia, precisão, sensitividade e especificidade, tabela 3. A precisão mostra que ambos modelos são capazes de classificar a classe COVID-19, porém, o modelo 15 se sobressai na classificação da classe Normal, ou seja, possui sensitividade significativamente maior se comparada a do modelo 49.

Tabela 3 – Comparação de métricas entre os modelos 15 e 49 - II.

Métrica	Modelo 15	Modelo 49
Acurácia	0,7439	0,6407
Precisão	0,9498	0,9933
Sensitividade	0,7252	0,6359
Especificidade	0,8279	0,8305

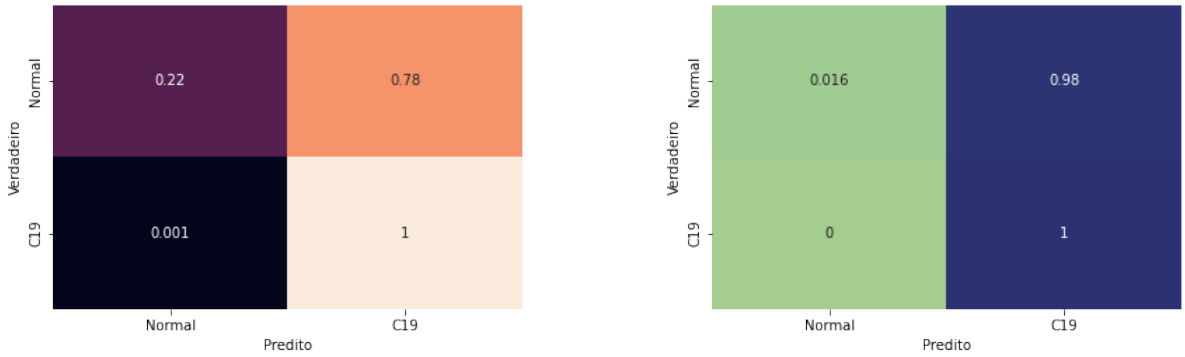
4.2 FASE DE TESTE

A fase de teste nos permite confirmar se a os resultados gerados pelo modelo são aceitáveis frente questões do mundo real, pois, imagens de origens diferentes às de treinamento são apresentadas ao modelo para avaliação de performance.

O teste é realizado utilizando subconjunto com 1.000 imagens: 500 para COVID-19 do banco hmHospitales, e 500 Normais do banco NIH. É importante ressaltar que as imagens que compõem a parcela de testes não foram expostas ao modelo na fase de treinamento.

As matrizes de confusão obtidas no subconjunto de teste para os modelos 15 e 49 estão presentes na figura 15. Novamente, é possível observar que ambos modelos possuem maior facilidade para classificar COVID-19 do que a classe Normal. Porém, agora, diferente do ocorrido no subconjunto de validação, nota-se que o poder de classificação do modelo 49 para a classe normal é ínfimo, em que apenas 1,6 % dos pulmões da classe Normal foram corretamente classificados.

Figura 15 – Matrizes de Confusão de Teste.



(a) Matriz de Confusão do Modelo 15. **(b) Matriz de Confusão do Modelo 50.**

As métricas advindas da matriz de confusão, Figura 15, estão relacionadas na Tabela 4.

Tabela 4 – Comparação de métricas entre os modelos 15 e 49 no subconjunto de teste.

Métrica	Modelo 15	Modelo 49
Acurácia	0,611	0,508
Precisão	0,999	1,0
Sensitividade	0,5625	0,5040
Especificidade	0,9955	1,0

Uma situação muito interessante se faz evidente ao se observar a tabela 4. Apesar do modelo 49 alcançar 1 nas métricas de precisão e especificidade, sua acurácia foi menor que a do modelo 15. Este fato se explica pelo fenômeno de sobreajuste, uma vez que este modelo é resultado da ultima época de treinamento e não possui a menor perda de validação. Sendo assim, é possível afirmar que o modelo 49 desenhou uma

fronteira de decisão que favorece a classe COVID-19 e ignora a classe Normal. Fato, este, que não é verdadeiro para o modelo 15, que possui a menor perda de validação e obteve um escore de acurácia 1,2 vezes maior que o modelo 49.

4.3 DISCUSSÃO

Os dois modelos possuem alto valor de especificidade, isto é, são raras as ocasiões em que um caso de COVID-19 será classificado como Normal. Porém, a margem entre falsos positivos do modelo 15 é de 20% comparado ao modelo 49. Sendo assim, o modelo que mais se adequá a uma possível implementação é o 15.

Apesar do modelo 15 ter classificado corretamente apenas 22% dos elementos da classe Normal na fase de teste, a métrica AP da fase de validação da classe normal, 0,7848, mostra que há margem para ajuste no limiar arbitrário que separa a atribuição de uma determinada saída na classe COVID-19 ou Normal, ou seja, ainda que a taxa de acerto demonstrada não seja ideal, com o mesmo modelo em mãos é possível se obter resultados taxas de acerto maiores.

A situação apresentada anteriormente não foi posta em prática pois seria necessária alterar a estrutura da aplicação Keras RetinaNet, o que se pretente fazer em trabalhos futuros.

O tempo de inferência para cada imagem de teste pelo modelo é de 0,48 segundos.

5 CONCLUSÕES

A pandemia de COVID-19 colocou os sistemas de saúde sob pressão. As técnicas mais utilizadas para detecção de COVID-19 são tomografia computadorizada, ultrassom dos pulmões e raio-X. Este último provém uma análise rápida de forma acessível, o que o torna uma alternativa factível para o sistema da saúde. Entretanto, na maior parte dos casos, os profissionais da medicina não possuem muita experiencia clínica. Mesmo que esta análise baseada em imagem seja aplicada, a acurácia do diagnóstico pode ser menor que a almejada em consequência da inexperiência e variabilidade de interpretação por estes profissionais.

Este trabalho apresenta uma metodologia prática para detectar e classificar pulmões por meio de imagens de raio-X de tórax entre a classe COVID-19 e Normal, munindo-se da arquitetura de detecção de um estágio. A metodologia, baseada na *RetinaNet*, abrange o problema tempo de execução e acurácia, explorando esta troca crucial e o refinando como um problema do mundo real e aplicável. Esta arquitetura de aprendizado profundo utiliza a *Focal Loss* e apresentou bons resultados com objetos densos, pequenos e desbalanceados.

No total, 3.000 imagens anotadas por profissionais da medicina foram consideradas: 2.000 compuseram a parcela de treinamento e 1.000 para validação. O resultado alcançado pela metodologia proposta na fase de teste implica em uma especificidade de 99%, precisão de 99%, e sensitividade de 56%. O escore alto de especificidade mostra que um paciente com COVID-19 dificilmente será classificado como Normal. E, apesar da taxa de acerto da classe Normal na fase de teste ser de 22%, a métrica AP da fase de validação da classe normal, 0.7848, demonstra que é possível, ainda com o mesmo modelo, alcançar resultados melhores para a classe normal. Para tanto, a aplicação Keras RetinaNet será modificada em trabalhos futuros.

Ainda em trabalhos futuros, o fluxo de trabalho pode ser expandido para dispositivos móveis, pois pode permitir o funcionamento da aplicação de maneira acessível no sistemas de saúde com um baixo custo de implementação.

Outra melhoria pode ser feita adicionando outras classes nos conjuntos de treino, validação e teste, uma vez que há outras doenças respiratórias que possuem semelhança na manifestação em imagens de raio-X, o que produziria uma fronteira de decisão mais adequada para o mundo real.

Por fim, o modelo pode ser avaliado pelas suas motivações que antecedem a tomada de decisão, como as regiões e detalhes que mais impactam na detecção e classificação do pulmão.

O presente trabalho foi publicado no congresso internacional *Latin American Conference on Computational Intelligence* através da IEEE sob o título *A practical Deep Learning approach to assist COVID-19 detection based on Chest X-ray images* (**camargo2021practicalC19**) - a ser publicado.

REFERÊNCIAS

ALBAWI, S.; MOHAMMED, T. A.; AL-ZAWI, S. Understanding of a convolutional neural network. In: IEEE. 2017 International Conference on Engineering and Technology (ICET). [*S. l.: s. n.*], 2017. P. 1–6.

BALTRUSCHAT, I. M. *et al.* Comparison of deep learning approaches for multi-label chest X-ray classification. **Scientific reports**, Nature Publishing Group, v. 9, n. 1, p. 1–10, 2019.

BHANDARY, A. *et al.* Deep-learning framework to detect lung abnormality–A study with chest X-Ray and lung CT scan images. **Pattern Recognition Letters**, Elsevier, v. 129, p. 271–278, 2020.

BOUREAU, Y.-L.; PONCE, J.; LECUN, Y. A theoretical analysis of feature pooling in visual recognition. In: PROCEEDINGS of the 27th international conference on machine learning (ICML-10). [*S. l.: s. n.*], 2010. P. 111–118.

BULLOCK, J.; CUESTA-LÁZARO, C.; QUERA-BOFARULL, A. XNet: A convolutional neural network (CNN) implementation for medical X-ray image segmentation suitable for small datasets. In: INTERNATIONAL SOCIETY FOR OPTICS e PHOTONICS. MEDICAL Imaging 2019: Biomedical Applications in Molecular, Structural, and Functional Imaging. [*S. l.: s. n.*], 2019. v. 10953, 109531z.

CHEN, J. *et al.* Foreground-background imbalance problem in deep object detectors: A review. In: IEEE. 2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR). [*S. l.: s. n.*], 2020. P. 285–290.

CHOLLET, F. **Deep learning with Python**. [*S. l.*]: Simon e Schuster, 2021.

CHOUHAN, V. *et al.* A novel transfer learning based approach for pneumonia detection in chest X-ray images. **Applied Sciences**, Multidisciplinary Digital Publishing Institute, v. 10, n. 2, p. 559, 2020.

CHOWDHURY, M. E. H. *et al.* Can AI Help in Screening Viral and COVID-19 Pneumonia? **IEEE Access**, v. 8, p. 132665–132676, 2020. DOI: 10.1109/ACCESS.2020.3010287.

CORONAVIRUS Pandemic (COVID-19) – the data. [*S. l.: s. n.*], 2021. Disponível em: https://ourworldindata.org/coronavirus-data?country=~BRA#increase-of-deaths.

CORONAVÍRUS: 8 gráficos para entender como a pandemia de covid-19 afetou as maiores economias do mundo. [*S. l.: s. n.*], 2021. Disponível em: https://www.bbc.com/portuguese/internacional-55835790.

GIRSHICK, R. Fast r-cnn. In: PROCEEDINGS of the IEEE international conference on computer vision. [*S. l.: s. n.*], 2015. P. 1440–1448.

GIRSHICK, R. *et al.* Rich feature hierarchies for accurate object detection and semantic segmentation. In: PROCEEDINGS of the IEEE conference on computer vision and pattern recognition. [*S. l.: s. n.*], 2014. P. 580–587.

HE, K. *et al.* Mask r-cnn. In: PROCEEDINGS of the IEEE international conference on computer vision. [*S. l.: s. n.*], 2017. P. 2961–2969.

HEMDAN, E. E.-D.; SHOUMAN, M. A.; KARAR, M. E. Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images. **arXiv preprint arXiv:2003.11055**, 2020.

HM Hospitales. Covid Data Save Lives. [*S. l.: s. n.*], 2020. Disponível em: https://www.hmhospitales.com/coronavirus/covid-data-save-lives/english-version.

HORNIK, K.; STINCHCOMBE, M.; WHITE, H. Multilayer feedforward networks are universal approximators. **Neural networks**, Elsevier, v. 2, n. 5, p. 359–366, 1989.

HUANG, R.; PEDOEEM, J.; CHEN, C. YOLO-LITE: a real-time object detection algorithm optimized for non-GPU computers. In: IEEE. 2018 IEEE International Conference on Big Data (Big Data). [*S. l.: s. n.*], 2018. P. 2503–2510.

ISMAEL, A. M.; ŞENGÜR, A. Deep learning approaches for COVID-19 detection based on chest X-ray images. **Expert Systems with Applications**, Elsevier, v. 164, p. 114054, 2021.

JUNGES, E. L. **Detecção e classificação de faltas em linhas de transmissão utilizando redes neurais artificiais**. 2018. B.S. thesis.

KANG, L. *et al.* Convolutional neural networks for document image classification. In: IEEE. 2014 22nd International Conference on Pattern Recognition. [*S. l.: s. n.*], 2014. P. 3168–3172.

KAUFMAN, S. *et al.* Leakage in data mining: Formulation, detection, and avoidance. **ACM Transactions on Knowledge Discovery from Data (TKDD)**, ACM New York, NY, USA, v. 6, n. 4, p. 1–21, 2012.

KESIM, E.; DOKUR, Z.; OLMEZ, T. X-ray chest image classification by a small-sized convolutional neural network. In: IEEE. 2019 scientific meeting on electrical-electronics & biomedical engineering and computer science (EBBT). [*S. l.: s. n.*], 2019. P. 1–5.

KHUZANI, A. Z.; HEIDARI, M.; SHARIATI, S. A. COVID-Classifier: An automated machine learning model to assist in the diagnosis of COVID-19 infection in chest x-ray images. **Scientific Reports**, Nature Publishing Group, v. 11, n. 1, p. 1–6, 2021.

KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. **arXiv preprint arXiv:1412.6980**, 2014.

KOITKA, S. *et al.* Ossification area localization in pediatric hand radiographs using deep neural networks for object detection. **PloS one**, Public Library of Science San Francisco, CA USA, v. 13, n. 11, e0207496, 2018.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. **Advances in neural information processing systems**, v. 25, p. 1097–1105, 2012.

LI, F.-F.; KARPATHY, A.; JOHNSON, J. Cs231n: Convolutional neural networks for visual recognition. **University lecture**, 2015.

LI, W. *et al.* Data augmentation for hyperspectral image classification with deep CNN. **IEEE Geoscience and Remote Sensing Letters**, IEEE, v. 16, n. 4, p. 593–597, 2018.

LI, X. *et al.* Multi-resolution convolutional networks for chest X-ray radiograph based lung nodule detection. **Artificial intelligence in medicine**, Elsevier, v. 103, p. 101744, 2020.

LIN, T.-Y.; DOLLÁR, P. *et al.* Feature pyramid networks for object detection. In: PROCEEDINGS of the IEEE conference on computer vision and pattern recognition. [*S. l.: s. n.*], 2017. P. 2117–2125.

LIN, T.-Y.; GOYAL, P. *et al.* Focal loss for dense object detection. In: PROCEEDINGS of the IEEE international conference on computer vision. [*S. l.: s. n.*], 2017. P. 2980–2988.

LIN, T.-Y.; MAIRE, M. *et al.* Microsoft coco: Common objects in context. In: SPRINGER. EUROPEAN conference on computer vision. [*S. l.: s. n.*], 2014. P. 740–755.

LIU, W. *et al.* Ssd: Single shot multibox detector. In: SPRINGER. EUROPEAN conference on computer vision. [*S. l.: s. n.*], 2016. P. 21–37.

MARTINS, R. d. A. *et al.* Contágio: história da prevenção das doenças transmissıveis. **São Paulo: Moderna**, p. 59–80, 1997.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, Springer, v. 5, n. 4, p. 115–133, 1943.

MESQUITA, M. E. R. V.; PIMENTA, M. A. Perceptrons Morfológico de Camada Única.

MINISTÉRIO da Saúde(BR). Secretaria de Ciência, Tecnologia, Inovação e Insumos Estratégicos em Saúde. Diretrizes para Diagnóstico e Tratamento da COVID-19. Brasília-DF. [*S. l.: s. n.*], 2020. Disponível em: https://portalarquivos.saude.gov.br/images/pdf/2020/May/08/Diretriz-Covid19-v4-07-05.20h05m.pdf.

NARIN, A.; KAYA, C.; PAMUK, Z. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. **Pattern Analysis and Applications**, Springer, p. 1–14, 2021.

OH, Y.; PARK, S.; YE, J. C. Deep learning covid-19 features on cxr using limited training data sets. **IEEE Transactions on Medical Imaging**, IEEE, v. 39, n. 8, p. 2688–2700, 2020.

OZTURK, T. *et al.* Automated detection of COVID-19 cases using deep neural networks with X-ray images. **Computers in biology and medicine**, Elsevier, v. 121, p. 103792, 2020.

PAN, S. J.; YANG, Q. A survey on transfer learning. **IEEE Transactions on knowledge and data engineering**, IEEE, v. 22, n. 10, p. 1345–1359, 2009.

PAPADIMITRIOU, P.; GARCIA-MOLINA, H. Data leakage detection. **IEEE Transactions on knowledge and data engineering**, IEEE, v. 23, n. 1, p. 51–63, 2010.

PARKHI, O. M.; VEDALDI, A.; ZISSERMAN, A. Deep face recognition. British Machine Vision Association, 2015.

PIRES, Í. R. N. *et al.* Um modelo estratégico para a análise de crédito utilizando redes neurais artificiais. Universidade Federal de Uberlândia, 2008.

RAHMAN, T. *et al.* Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images. **Computers in Biology and Medicine**, v. 132, p. 104319, 2021. ISSN 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2021.104319. Disponível em: https://www.sciencedirect.com/science/article/pii/S001048252100113X.

REN, S. *et al.* Faster r-cnn: Towards real-time object detection with region proposal networks. **Advances in neural information processing systems**, v. 28, p. 91–99, 2015.

ROSENBLATT, F. **The perceptron, a perceiving and recognizing automaton Project Para**. [*S. l.*]: Cornell Aeronautical Laboratory, 1957.

ROY, S. *et al.* Deep learning for classification and localization of COVID-19 markers in point-of-care lung ultrasound. **IEEE Transactions on Medical Imaging**, IEEE, v. 39, n. 8, p. 2676–2687, 2020.

RUDER, S. An overview of gradient descent optimization algorithms. **arXiv preprint arXiv:1609.04747**, 2016.

RUSSAKOVSKY, O. *et al.* ImageNet Large Scale Visual Recognition Challenge. **International Journal of Computer Vision (IJCV)**, v. 115, n. 3, p. 211–252, 2015. DOI: 10.1007/s11263-015-0816-y.

SAIZ, F. A.; BARANDIARAN, I. COVID-19 Detection in Chest X-ray Images using a Deep Learning Approach. **Int. J. Interact. Multim. Artif. Intell.**, v. 6, n. 2, p. 1–4, 2020.

SAMMUT, C.; WEBB, G. I. **Encyclopedia of machine learning**. [*S. l.*]: Springer Science & Business Media, 2011.

SANTOS, A. M. d. *et al.* Usando redes neurais artificiais e regressão logıstica na predição da hepatite A. **Revista Brasileira de Epidemiologia**, SciELO Brasil, v. 8, p. 117–126, 2005.

SHORTEN, C.; KHOSHGOFTAAR, T. M. A survey on image data augmentation for deep learning. **Journal of Big Data**, Springer, v. 6, n. 1, p. 1–48, 2019.

SILVA, A. Utilização de Redes Neurais Artificiais para Classificação de SPAM. **Masters, Centro Federal De Educaçao Tecnológica De Minas Gerais**, 2009.

SUN, L. *et al.* High-order feature learning for multi-atlas based label fusion: Application to brain segmentation with MRI. **IEEE Transactions on Image Processing**, IEEE, v. 29, p. 2702–2713, 2019.

SUTSKEVER, I. *et al.* On the importance of initialization and momentum in deep learning. In: PMLR. INTERNATIONAL conference on machine learning. [*S. l.: s. n.*], 2013. P. 1139–1147.

TAN, M.; LE, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In: PMLR. INTERNATIONAL Conference on Machine Learning. [*S. l.: s. n.*], 2019. P. 6105–6114.

TAQI, A. M. *et al.* The impact of multi-optimizers and data augmentation on TensorFlow convolutional neural network performance. In: IEEE. 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR). [*S. l.: s. n.*], 2018. P. 140–145.

TIAN, H. *et al.* An investigation of transmission control measures during the first 50 days of the COVID-19 epidemic in China. **Science**, American Association for the Advancement of Science, v. 368, n. 6491, p. 638–642, 2020.

TORREY, L.; SHAVLIK, J. Transfer learning. In: HANDBOOK of research on machine learning applications and trends: algorithms, methods, and techniques. [*S. l.*]: IGI global, 2010. P. 242–264.

WANG, S.; YANG, D. M. *et al.* Pathology image analysis using segmentation deep learning algorithms. **The American journal of pathology**, Elsevier, v. 189, n. 9, p. 1686–1698, 2019.

WANG, S.; ZHA, Y. *et al.* A fully automatic deep learning system for COVID-19 diagnostic and prognostic analysis. **European Respiratory Journal**, Eur Respiratory Soc, v. 56, n. 2, 2020.

WANG, T. *et al.* End-to-end text recognition with convolutional neural networks. In: IEEE. PROCEEDINGS of the 21st international conference on pattern recognition (ICPR2012). [*S. l.: s. n.*], 2012. P. 3304–3308.

WANG, X. *et al.* Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In:

PROCEEDINGS of the IEEE conference on computer vision and pattern recognition. [*S. l.: s. n.*], 2017. P. 2097–2106.

WEISS, K.; KHOSHGOFTAAR, T. M.; WANG, D. A survey of transfer learning. **Journal of Big data**, SpringerOpen, v. 3, n. 1, p. 1–40, 2016.

WORLD Health Organization. Coronavirus disease (COVID-19) Pandemic. [*S. l.: s. n.*], 2020. Disponível em: https://www.who.int/emergencies/diseases/novel-coronavirus-2019.

YADAV, S. S.; JADHAV, S. M. Deep convolutional neural network based medical image classification for disease diagnosis. **Journal of Big Data**, Springer, v. 6, n. 1, p. 1–18, 2019.

ZHANG, K. *et al.* Clinically applicable AI system for accurate diagnosis, quantitative measurements, and prognosis of COVID-19 pneumonia using computed tomography. **Cell**, Elsevier, v. 181, n. 6, p. 1423–1433, 2020.

ÍNDICE REMISSIVO