



MONOGRAFIA DE PROJETO FINAL DE GRADUAÇÃO

**REDES NEURAIS CONVOLUCIONAIS
E APRENDIZADO POR MÚLTIPLAS INSTÂNCIAS PARA CLASSIFICAÇÃO
DE CÂNCER DE MAMA**

Eduardo Augusto Crozarioli Lopes Filho

Hugo Silva de Vasconcelos

Curso Superior de Engenharia de Redes de Comunicação

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

UNIVERSIDADE DE BRASÍLIA

Faculdade de Tecnologia

MONOGRAFIA DE PROJETO FINAL DE GRADUAÇÃO

**REDES NEURAIS CONVOLUCIONAIS
E APRENDIZADO POR MÚLTIPLAS INSTÂNCIAS PARA CLASSIFICAÇÃO
DE CÂNCER DE MAMA**

Eduardo Augusto Crozarioli Lopes Filho

Hugo Silva de Vasconcelos

*Monografia de Projeto Final de Graduação submetida ao Departamento
de Engenharia Elétrica como requisito parcial para obtenção do grau de
Bacharel em Engenharia de Redes de Comunicação*

Banca Examinadora

Prof. Dr. Daniel Guerreiro e Silva, EnE/UnB

Orientador

Prof. Dr. Georges Daniel Amvame Nze, EnE/UnB

Examinador Interno

Prof. Dr. Alexandre Ricardo Soares Romariz,

EnE/UnB

Examinador Interno

FICHA CATALOGRÁFICA

LOPES FILHO, EDUARDO AUGUSTO CROZARIOLI

DE VASCONCELOS, HUGO SILVA

REDES NEURAIIS CONVOLUCIONAIS E APRENDIZADO POR MÚLTIPLAS INSTÂNCIAS PARA CLASSIFICAÇÃODE CÂNCER DE MAMA [Distrito Federal] 2023.

xvi, 47 p., 210 x 297 mm (ENE/FT/UnB, Bacharel, Engenharia de Redes de Comunicação, 2023).

Monografia de Projeto Final de Graduação - Universidade de Brasília, Faculdade de Tecnologia.

Departamento de Engenharia Elétrica

1. Breast Cancer Classification

2. Multiple Instance Learning

3. Convolutional Neural Network

4. Deep Learning

I. ENE/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

LOPES FILHO, E.A.C; DE VASCONCELOS, H.S. (2023). *REDES NEURAIIS CONVOLUCIONAIS E APRENDIZADO POR MÚLTIPLAS INSTÂNCIAS PARA CLASSIFICAÇÃODE CÂNCER DE MAMA*.

Monografia de Projeto Final de Graduação, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 47 p.

CESSÃO DE DIREITOS

AUTORES: Eduardo Augusto Crozarioli Lopes Filho; Hugo Silva de Vasconcelos

TÍTULO: REDES NEURAIIS CONVOLUCIONAIS E APRENDIZADO POR MÚLTIPLAS INSTÂNCIAS PARA CLASSIFICAÇÃODE CÂNCER DE MAMA.

GRAU: Bacharel em Engenharia de Redes de Comunicação

ANO: 2023

É concedida à Universidade de Brasília permissão para reproduzir cópias desta Monografia de Graduação e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. Do mesmo modo, a Universidade de Brasília tem permissão para divulgar este documento em biblioteca virtual, em formato que permita o acesso via redes de comunicação e a reprodução de cópias, desde que protegida a integridade do conteúdo dessas cópias e proibido o acesso a partes isoladas desse conteúdo. O autor reserva outros direitos de publicação e nenhuma parte deste documento pode ser reproduzida sem a autorização por escrito do autor.

Eduardo Augusto Crozarioli Lopes Filho
Depto. de Engenharia Elétrica (ENE) - FT
Universidade de Brasília (UnB)
Campus Darcy Ribeiro
CEP: 70919-970 - Brasília-DF - Brasil

Hugo Silva de Vasconcelos
Depto. de Engenharia Elétrica (ENE) - FT
Universidade de Brasília (UnB)
Campus Darcy Ribeiro
CEP: 70919-970 - Brasília-DF - Brasil

Aos meus pais, que sempre fizeram o possível e o impossível por mim - Eduardo Augusto Crozarioli Lopes Filho

À mulher extraordinária que é minha mãe, que enfrentou inúmeras batalhas para pavimentar o caminho que me trouxe até este momento - Hugo Silva de Vasconcelos

AGRADECIMENTOS

Durante minha jornada acadêmica, muitas pessoas foram essenciais para o meu crescimento pessoal e profissional. Gostaria de expressar minha gratidão à minha família, que sempre esteve ao meu lado, oferecendo apoio nos momentos difíceis e celebrando minhas conquistas. Em especial gostaria de agradecer à minha mãe Helena e ao meu pai Eduardo, que sempre fizeram o possível e o impossível por mim.

Agradeço também aos meus amigos, que desempenharam um papel fundamental ao longo da minha graduação, transformando momentos de dificuldades em lembranças boas. Agradeço ao Hugo pelos mais de dez anos de amizade. Crescemos juntos, nos formamos juntos, trabalhamos juntos e estamos novamente finalizando mais um ciclo juntos. Júlia, Nina e Klainer, amigos especiais que a Engenharia de Redes me deu. Ao lado de vocês vivi tantos momentos bons que é fácil perder a conta.

Por fim, gostaria de agradecer aos meus professores e professoras, que sempre desempenharam um papel fundamental na minha formação. Se hoje sou capaz de superar dificuldades tanto dentro da Universidade quando no Trabalho foi graças ao incentivo de vocês.

Eduardo Augusto Crozarioli Lopes Filho

Agradeço à minha mãe Claudeci, minha tia Carla e minha avó Maria por sempre me incentivarem nos estudos, me apoiarem nas minhas decisões e por todo amor e carinho.

Aos meus amigos que a Engenharia de Redes me proporcionou, em especial, Eduardo, Júlia, Klainer e Sabrina que proporcionaram momentos inesquecíveis durante o curso.

Aos professores da Universidade de Brasília que fizeram parte da minha trajetória, em especial ao professor Daniel Guerreiro que além de professor e orientador, também foi um ótimo conselheiro quando precisei.

Finalmente, mas igualmente significativo gostaria de agradecer a toda minha família, pelo apoio, motivação e incentivo a seguir os meus sonhos.

Hugo Silva de Vasconcelos

RESUMO

Estudos realizados pelo Instituto Nacional do Câncer apontam que, durante o triênio entre 2023 e 2025, cerca de 704 mil novos casos de câncer surgirão no Brasil, sendo o câncer de mama o mais comum entre as mulheres. Detectar o câncer de mama em estágios iniciais, e principalmente, determinar seu status de receptores hormonais (Estrogênio e/ou Progesterona) é um ponto chave para definir o melhor tratamento. O método atual para determinar o status de receptores hormonais em pacientes com câncer é por meio do exame Imuno-Histoquímico, um exame complexo que exige a análise profunda de um patologista. O propósito deste trabalho é apresentar um modelo de Rede Neural Convolutacional Profunda capaz de determinar, a partir de exames de imagem, o Status de Receptores de Estrogênio em pacientes com câncer de mama, oferecendo uma alternativa mais ágil quando comparada ao método convencional de classificação de câncer de mama, auxiliando na análise realizada por patologistas e facilitando o processo de diagnóstico. A ferramenta desenvolvida foi treinada utilizando dados clínicos de 1093 pacientes e 1888 exames de imagem, obtendo uma AUC de 0.82.

Palavras-chave: Câncer de Mama, Redes Neurais Profundas, Redes Neurais Convolucionais , Aprendizado por Múltiplas Instâncias, ER-Status

ABSTRACT

Studies conducted by the National Cancer Institute indicate that, during the triennium between 2023 and 2025, approximately 704 thousand new cases of cancer will emerge in Brazil, with breast cancer being the most common among women. Detecting breast cancer in its early stages, and particularly determining its hormonal receptor status (Estrogen and/or Progesterone), is a key factor in defining the best treatment. The current method for determining hormonal receptor status in cancer patients is through Immunohistochemical examination, a complex test that requires in-depth analysis by a pathologist. The purpose of this work is to present a model of a Deep Convolutional Neural Network capable of determining, from image exams, the Estrogen Receptor Status in breast cancer patients, providing a more efficient alternative compared to the conventional method of breast cancer classification. This assists in the analysis performed by pathologists and facilitates the diagnostic process. The developed tool was trained using clinical data from 1093 patients and 1888 image exams, achieving an AUC of 0.82.

Palavras-chave: Breast Cancer, Deep Neural Networks, Convolutional Neural Networks , Multiple Instance Learning, ER-Status

SUMÁRIO

1	INTRODUÇÃO	1
1.1	CONTEXTUALIZAÇÃO	1
1.2	PROBLEMÁTICA	2
1.3	OBJETIVOS	3
1.4	ESTRUTURA DOCUMENTAL	4
2	FUNDAMENTAÇÃO TEÓRICA	5
2.1	REDES NEURAIS ARTIFICIAIS	5
2.1.1	PERCEPTRON	6
2.1.2	FUNÇÕES DE ATIVAÇÃO	6
2.1.3	TREINAMENTO	7
2.2	REDES NEURAIS CONVOLUCIONAIS	8
2.2.1	CAMADAS CONVOLUCIONAIS	10
2.2.2	CAMADAS DE POOLING	11
2.2.3	CAMADAS DENSAS	13
2.3	APRENDIZADO POR MÚLTIPLAS INSTÂNCIAS	14
2.4	REDES RESIDUAIS	15
2.5	MÉTODOS DE AVALIAÇÃO	18
2.5.1	MATRIZ DE CONFUSÃO	18
2.5.2	PRECISÃO	19
2.5.3	REVOCAÇÃO OU <i>Recall</i>	19
2.5.4	<i>F1-Score</i>	20
2.5.5	CURVA ROC E ÁREA SOB A CURVA (AUC)	20
3	METODOLOGIA	22
3.1	OBTENÇÃO DOS DADOS	22
3.1.1	CRIANDO ARQUIVO DE MANIFESTO	22
3.1.2	UTILIZANDO O ARQUIVO DE MANIFESTO	24
3.2	PRÉ-PROCESSAMENTO E TRATAMENTO DOS DADOS	25
3.2.1	MÉTODO DE OTSU E REMOÇÃO DE PLANO DE FUNDO	25
3.2.2	FRAGMENTAÇÃO/TILLING	27
3.2.3	BALANCEAMENTO DOS DADOS	28
3.2.4	DESCONTAMINAÇÃO DOS DADOS	29
3.3	ARQUITETURA DA REDE	30
3.4	TREINAMENTO	33
4	RESULTADOS E DISCUSSÕES	36
4.1	REDE DE CONTROLE	36

4.2	RECEPTORNET SIMPLIFICADA	39
5	CONCLUSÃO E TRABALHOS FUTUROS	44
	REFERÊNCIAS BIBLIOGRÁFICAS	46

LISTA DE FIGURAS

2.1	Rede MLP.....	5
2.2	Perceptron.....	6
2.3	Comparativo da densidade de conexões entre redes MLP e redes convolucionais. Em (a), cada neurônio de entrada X_n está diretamente conectado aos neurônios de saída S_n , criando uma rede Totalmente Conectada. Enquanto em (b), os neurônios de entrada X_n estão conectados apenas com alguns neurônios de saída S_n , criando uma rede com conexões esparsas.	9
2.4	Arquitetura da LeNet, a primeira Rede Neural Convolucional implementada. FONTE: ZHANG, Aston & ZACHARY, Lipton. - Dive into Deep Learning	9
2.5	Conexão Esparsa: Nem todos os neurônios estão conectados diretamente entre si, porém, mesmo em camadas distantes, o Neurônio g_3 é influenciado pelas entradas x_n , uma vez que estão indiretamente conectados.....	10
2.6	Mapa de Atributos: Convolução entre uma entrada 28x28 e Kernel 5x5, criando na saída um mapa de atributos de dimensão 24x24. FONTE: NIELSEN, Michael A. - Neural Networks and Deep Learning.	11
2.7	<i>Pooling (MaxPooling)</i> aplicado em um Mapa de Atributos. A fileira inferior representa a saída da Camada Convolucional, enquanto a fileira superior representa a saída da Camada de <i>Pooling</i> . Na Figura (a) é possível observar que o valor do pixel após Camada de <i>Pooling</i> representa o valor máximo entre os pixels central e seus vizinhos. Na Figura (b), o Mapa de Atributos é transladado para direita. Mesmo após o deslocamento, é possível observar que a saída da camada de <i>pooling</i> é igual a vista em (a), porém, transladada para direita.	12
2.8	Max Pooling aplicado à uma Mapa de Atributos 4x4	13
2.9	Na Figura (a), é possível observar que, a partir de 2012, com a AlexNet, as Redes Convolucionais se sagraram campeãs do desafio. Além disso, nota-se que a ResNet obteve uma acurácia superior a classificação realizada por seres humanos. A Figura (b) mostra a quantidade de camadas utilizadas nos modelos vencedores em cada ano.....	16
2.10	Bloco Residual de uma ResNet.	17
2.11	Matriz de Confusão	19
2.12	Área sob a curva ROC (AUC), FONTE: developers.google.com.....	21
3.1	Base de Dados do TGCA-BRCA.	23
3.2	Base de Dados do TGCA-BRCA.	24
3.3	Slide de Tecido: A imagem apresenta um slide completo com duas amostras de tecido. O plano de fundo (branco) faz parte do slide original e corresponde a mais da metade da imagem.	26
3.4	Exemplo de quatro <i>tiles</i> válidos, gerados a partir de um Slide Inteiro.....	26
3.5	Método de Otsu aplicado nos <i>Tiles</i> . O Método tingiu de amarelo todos os pixels classificados como background, enquanto os pixels que formam o tecido foram tingidos de roxo.....	27

3.6	Slide Reconstruído: A Figura apresenta imagem reconstruída a partir dos <i>tiles</i> com pelo menos 1% de informação. O algoritmo criado substitui os <i>tiles</i> inexistentes (<i>background</i>) por <i>tiles</i> pretos.....	28
3.7	Esquemático do Modelo	30
3.8	Módulo de Extração de Características.....	31
3.9	Módulo de Atenção	32
3.10	Módulo de Decisão	33
4.1	Curvas de Custo (<i>loss</i>), Erro (<i>error</i>), Taxa de Falsos Positivo (FPR) e Taxa de Falsos Negativo (FNR) a cada época de treinamento do modelo de 30 épocas	37
4.2	Matriz de Confusão da Rede Neural Convolucional Convencional	37
4.3	Curva ROC.....	38
4.4	Curvas de Custo (<i>loss</i>), Erro (<i>error</i>), Taxa de Falsos Positivo (FPR) e Taxa de Falsos Negativo (FNR) a cada época de treinamento do modelo de 105 épocas.....	39
4.5	Matriz de Confusão da Rede Neural Convolucional Convencional	39
4.6	Curvas de Custo (<i>loss</i>), Erro (<i>error</i>), Taxa de Falsos Positivo (FPR) e Taxa de Falsos Negativo (FNR) a cada época de treinamento do modelo final treinado por 50 épocas	40
4.7	Matriz de Confusão da ReceptorNet Simplificada	41
4.8	Curva ROC.....	41
4.9	Curvas de Perda (<i>loss</i>), Erro (<i>error</i>), Taxa de Falsos Positivo (FPR) e Taxa de Falsos Negativo (FNR) a cada época de treinamento do modelo final treinado por 100 épocas	42
4.10	Matriz de Confusão da ReceptorNet Simplificada	42
4.11	Curva ROC.....	43

LISTA DE TABELAS

3.1	Quantidade de Slides em cada Conjunto de Dados.....	34
3.2	Comparação entre os Níveis dos Slides e Resolução correspondente. A Resolução em cada nível varia para cada Slide diferente, porém, os valores apresentados foram obtidos por meio de cálculo de média simples entre 10 Slides aleatórios.	35

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

Segundo estudos realizados pelo Instituto Nacional do Câncer (INCA), durante o triênio de 2023 e 2025, são esperados 704 mil novos casos de câncer no Brasil, sendo o câncer de mama, o segundo com maior incidência do país e o mais comum entre as mulheres [1]. Para este tipo de câncer, um diagnóstico antecipado e efetivo pode ser a diferença entre um tratamento mais brando e com altas taxas de recuperação ou opções de tratamentos mais invasivos e perigosos. Detectar o câncer de mama, e principalmente, determinar seus status de receptores hormonais (Estrogênio e Progesterona) é um ponto chave para definir um prognóstico e decidir por determinado tipo de tratamento [2]. Grande parte dos Cânceres de Mama são Receptores Hormonais Positivos, ou seja, as células cancerígenas crescem na presença de Estrogênio(ER) e/ou Progesterona(PR). Nestes casos da doença, a detecção do status do receptor é fundamental para proporcionar um tratamento mais eficaz. De acordo com Portais Informativos sobre Câncer e Hospitais especializados no combate ao câncer, a Hormonioterapia é comumente indicada para o câncer de mama com células que apresentam positividade para receptores hormonais, uma vez que, este tipo de tratamento reduz os níveis de Estrogênio/Progesterona ou bloqueiam os receptores de Estrogênio/Progesterona, impedindo o crescimento de novas células cancerígenas [3].

Antes de se tornar uma área da computação capaz de resolver problemas complexos nos mais variados setores da sociedade, a Inteligência Artificial (IA) passou por alguns "invernos", desde quando foi proposta pela primeira vez em 1955 [4]. Estes invernos foram períodos onde o financiamento para pesquisas e o interesse sobre a área da IA diminuíram drasticamente, devido às barreiras tecnológicas e restrições do próprio modelo criado na época. Seja por ainda não existir um hardware otimizado o suficiente para o processamento de grandes matrizes, seja pelas limitações do modelo de Rede Neural de Camada Única proposto inicialmente, que ainda era incapaz de solucionar uma simples não linearidade como uma porta XOR ou pela inexistência de dados o suficiente para realizar o treinamento adequado de um modelo de rede neural mais complexa, essa área da computação permaneceu estagnada por décadas até que estes problemas fossem superados.

Desde os anos 2000 estamos vivendo um "verão" na área da IA. Com o surgimento de Placas Gráficas especializadas no processamento de imagem, e também matrizes, superando a restrição de uma única camada oculta e desenvolvendo Redes Neurais de Múltiplas Camadas, além das grandes quantidades de dados que são produzidos diariamente, devido a popularização de computadores pessoais e smartphones, a Inteligência Artificial retomou o protagonismo e se tornou uma das áreas de tecnologia que mais cresce, sendo na medicina, uma de suas principais aplicações.

A Medicina Baseada em Evidência (MBE), também citada como *Scientific Medicine*, é definida como o uso consciente, explícito e criterioso das melhores evidências científicas existentes, somado à experiência individual clínica na tomada de decisões sobre o cuidado de pacientes [5]. Apesar da metodologia científica ser aplicada há muitos anos na medicina, o termo *Evidence-Based Medicine* surgiu pela primeira vez em 1992, quando o Dr. Gordon Guyatt publicou no *American College of Physicians Journal Club* um programa

de Residência intitulado com esse nome [6][7]. Desde seu surgimento, umas das maiores contribuições da MBE foi desenvolver de forma sistemática, a produção e revisão de dados clínicos. Em outras palavras, priorizou-se ainda mais os dados clínicos anteriores para as tomadas de decisões, criando-se bases de dados cada vez mais completas e organizadas [8].

Um elo em comum, e que conecta a Inteligência Artificial e a Medicina Baseada em Evidências é a importância da utilização de dados passados para a tomada de decisões. Devido o crescimento no número de dados clínicos disponíveis atualmente, tornou-se viável treinar redes neurais profundas para gerar diagnósticos médicos com precisão. Estudos mostram [9][10][11][12] que a utilização de Redes Neurais Convolucionais para Reconhecimento de Padrões em Exames de Imagem é uma opção eficaz e capaz de detectar e até mesmo prever diferentes tipos de câncer.

Neste trabalho, foi desenvolvido um Modelo de Rede Neural Convolucional Profunda capaz de determinar o Status de Receptores Hormonais em pacientes com Câncer de Mama utilizando diretamente Exames de Imagem como coloração H&E.

1.2 PROBLEMÁTICA

Durante o tratamento de câncer, um dos pontos principais e que aumentam consideravelmente as chances de sucesso, é um diagnóstico precoce da doença [13][14]. Detectar o tipo e subtipo de câncer em seus estágios iniciais permite um tratamento mais assertivo e tranquilo com maiores taxas de recuperação. Atualmente, o procedimento mais comum para determinar o status de receptores hormonais em pacientes com câncer de mama é o Exame Imuno-Histoquímico (IHC). Para realizar este exame, uma biopsia do tecido a ser analisado é separada e submetida a uma sequência de processos químicos [15]. Os principais passos para realização do IHC são:

1. Desparafinização e Hidratação: Fase inicial do exame IHC, onde o tecido higienizado e preparado para as etapas subsequentes. A amostra é submersa em diferentes concentrações de Xileno e Álcool.
2. Recuperação de Antígeno: Durante o estágio de Recuperação de Antígeno, o tecido é submetido ao método de *Heat-Induced Antigen Retrieval (HIAR)*, onde ele é aquecido em água quente, micro ondas, autoclave ou forno, para a evaporação do formol e outros conservantes adicionados na amostra. Esse procedimento facilita a reação da amostra com os anticorpos que serão acrescentados nas etapas seguintes.
3. Adição de Anticorpos: Por fim, são adicionados os anticorpos à amostra. Esta é a etapa principal do IHC, onde os anticorpos reagem com os antígenos presentes no tecido analisado. Essa reação entre tecido e anticorpos alteram a coloração da amostra, possibilitando sua análise por meio de um microscópio de luz, e futuramente, a classificação do status dos receptores hormonais do paciente por um patologista. A quantidade e concentração dos anticorpos utilizados nessa etapa depende tanto do laboratório que está realizando o exame quanto do tecido que está sendo submetido ao exame.

O IHC, apesar de muito utilizado para classificação de Câncer de Mama, é um exame com grande variabilidade em seu resultado final. A amostra de tecido cancerígeno normalmente é armazenada por

períodos diferentes de tempo até chegar ao laboratório em que será processada, sendo necessário adaptar a etapa de higienização para cada amostra. Além disso, a concentração e a quantidade de anticorpos diferentes que serão adicionados ao tecido durante a realização do exame também variam de acordo com as condições do tecido ou o tipo de câncer que está sendo investigado. Por fim, a amostra final passa pelo diagnóstico subjetivo de um patologista, que analisa a amostra de forma visual.

A *American Society of Clinical Oncology (ASCO)* em conjunto com a *College of American Pathologists (CAP)* publicaram um artigo[16] com Diretrizes para realizar o Exame Imuno-Histoquímico. O artigo aborda de forma abrangente recomendações referente a realização do Exame IHC, apresentando detalhes sobre o período ideal de armazenamento da amostra, manuseio, protocolo de testagem e até recomendações referentes ao credenciamento dos laboratórios onde o exame será realizado. Por fim, a ASCO/CAP estima que 20% dos exames imuno-histoquímicos realizados atualmente sejam imprecisos, contendo falso-negativos e falso-positivos.

Por outro lado, a Coloração Hematoxilina-Eosina (*H&E Staining*)[17][18] é o método de coloração de tecido mais utilizado na histologia. Esta técnica permite analisar visualmente a morfologia do tecido, a partir de reações químicas entre a amostra, a Hematoxilina e a Eosina. As etapas principais [17] da Coloração H&E podem ser observadas abaixo:

1. Desparafinização e Hidratação: Assim como no IHC, a amostra de tecido deve ser preparada higienizada e preparada. A amostra é submersa em diferentes concentrações de Xileno e Álcool.
2. Hematoxilina: Um solução de Hematoxilina é adicionada à amostras. Por ser um corante básicos, estruturas celulares ácidas, como os núcleos, reagem com a hematoxilina, passando a ter uma tonalidade azul-arroxeadado [18].
3. Higienização: Etapa de higienização entre pigmentações, onde a solução aplicada anteriormente é removida. O procedimento é simples e normalmente a amostra é colocada em água corrente.
4. Eosina: Um solução de Eosina é adicionada à amostras. Por ser um corante ácido, estruturas celulares básicas, como citoplasma e mitocôndrias, reagem com a eosina, passando a ter uma tonalidade rosa-avermelhado [18].

Pelo fato de tingir estruturas celulares diferentes com tonalidades diferentes, a coloração H&E permite observar a morfologia do tecido analisado. Além disso, por ser um método de coloração de tecidos muito difundido, o H&E é o ponto de partida para muitos exames, inclusive o próprio Exame Imuno-Histoquímico. A popularidade deste método também reflete na quantidade de Exames de Imagens disponíveis com esse padrão, o que favorece o treinamento de uma Rede Neural Profunda.

1.3 OBJETIVOS

Este trabalho tem como objetivo principal criar um modelo de Rede Neural Profunda, baseado na ReceptorNet [9], capaz de determinar o Status de Receptor de Estrogênio (ER-Status) a partir de Exames

de Imagem com Coloração H&E em pacientes com Câncer de Mama. A ReceptorNet é um modelo robusto e que comprovou a viabilidade de se treinar uma Rede Neural Profunda para esta tarefa de classificação, porém, exige uma infraestrutura poderosa para isso. Será apresentado um modelo simplificado, capaz de determinar o ER-Status, utilizando uma infraestrutura reduzida.

1.4 ESTRUTURA DOCUMENTAL

Este documento foi dividido em cinco capítulos principais. Inicialmente, o Capítulo de Introdução traz uma contextualização sobre o tema e apresenta os Objetivos que guiam este trabalho. Em seguida, no Capítulo de Fundamentação Teórica serão abordados os conceitos teóricos necessários para compreensão do trabalho realizado. No Capítulo de Metodologia serão detalhados os procedimentos e abordagens adotados durante o desenvolvimento desse trabalho. No Capítulo de Resultados e Discussões serão apresentadas métricas obtidos durante o treinamento da rede e Discussões que envolvem esses resultados. Por fim, o Capítulo de Conclusão traz as considerações finais referente a este trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, serão expostos os fundamentos teóricos essenciais para a compreensão deste trabalho. A Seção se inicia com uma introdução às Redes Neurais Artificiais e Redes Neurais Convolucionais. Em seguida, será introduzido o Modelo de Aprendizado por Múltiplas Instâncias (MIL) e suas aplicações específicas no âmbito da medicina. Será explorado como esse modelo é utilizado para lidar com conjuntos de instâncias, destacando sua relevância em tarefas de classificação de exames de imagem. Logo após, serão abordadas as Redes Residuais e sua relevância na concepção de Redes Convolucionais cada vez mais profundas. Serão apresentados resultados obtidos por redes cada vez mais profundas em tarefas de classificação e como a implementação de conexões residuais revolucionou as redes neurais convolucionais. Por fim, a seção se encerra trazendo as principais métricas utilizadas para a avaliação de redes neurais.

2.1 REDES NEURAS ARTIFICIAIS

As Redes Neurais Artificiais (RNAs) representam uma classe de algoritmos de aprendizado de máquina inspirados na organização do cérebro humano, que é composto por bilhões de neurônios interconectados. Nas RNAs, neurônios artificiais são organizados em camadas com diferentes arquiteturas, adaptadas às necessidades específicas da aplicação. Entre os tipos de RNAs, o modelo mais proeminente é o *Multilayer Perceptron* (MLP), caracterizado pela presença de perceptrons organizados em camadas distintas, incluindo uma camada de entrada, múltiplas camadas ocultas e uma camada de saída, como se observa na Figura 2.1, onde a camada azul representa a camada de entrada, as camadas verdes representam as camadas ocultas e a camada laranja representa a saída.

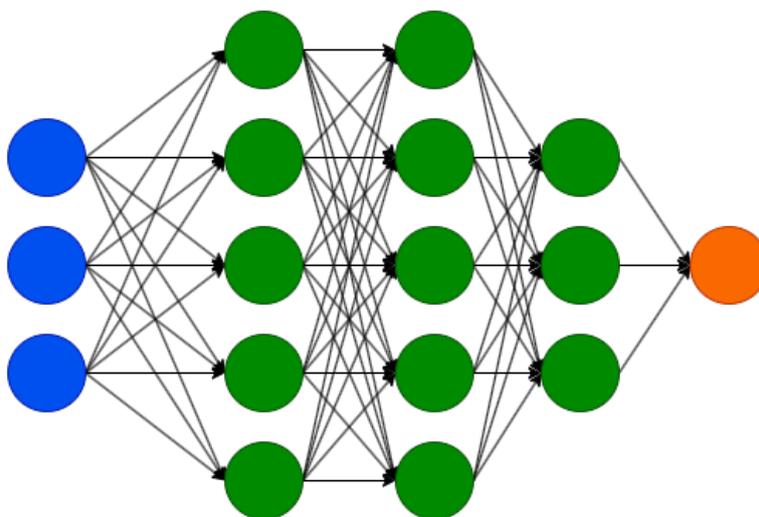


Figura 2.1: Rede MLP

2.1.1 Perceptron

Os perceptrons, analogamente aos neurônios biológicos, desempenham um papel crucial na MLP, contribuindo para o processamento global da informação. O processamento da informação dentro de cada neurônio é dado pela equação

$$y = f(\omega^T x + \omega_0), \quad (2.1)$$

onde x representa um vetor de entrada que realiza um produto escalar com um vetor ω de pesos e soma-se a um termo de bias ω_0 e em seguida, aplica o resultado em uma função de ativação f para gerar uma saída y [19]. Na Figura 2.2 é possível observar que o vetor de entradas x é composto por x_0, x_1 e x_2 e o vetor de pesos ω é composto por $\omega_0, \omega_1, \omega_2$, onde x_0 e ω_0 juntos compõem o termo de bias. Embora um único perceptron seja intrinsecamente limitado a aprender funções lineares, sua capacidade de inferência pode ser ampliada quando vários perceptrons são combinados em camadas.

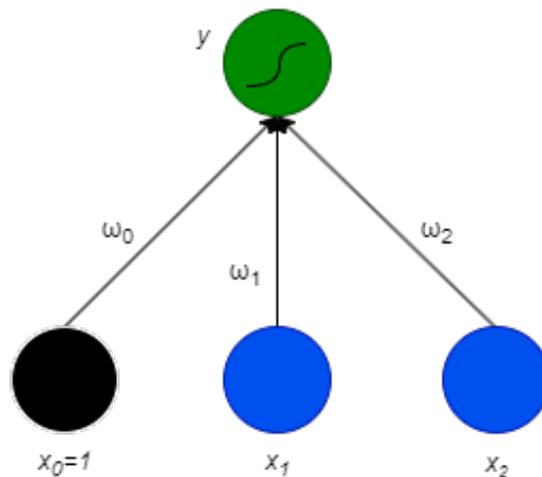


Figura 2.2: Perceptron

2.1.2 Funções de Ativação

Para expandir a capacidade de inferência da RNA, diferentes funções de ativação podem ser atribuídas a cada perceptron. Essas funções são fundamentais para introduzir não linearidades no modelo da RNA, permitindo-lhe capturar relacionamentos complexos nos dados. Alguns exemplos de funções de ativação incluem a função sigmoide,

$$\text{sigmoid}(\omega, x) = \frac{1}{1 + e^{-(\omega^T x + \omega_0)}}, \quad (2.2)$$

e a função ReLU (Unidade Linear Retificada),

$$\text{ReLU}(\omega, x) = \begin{cases} 0, & \text{se } (\omega^T x + \omega_0) \leq 0 \\ \omega^T x + \omega_0, & \text{caso o contrário.} \end{cases} \quad (2.3)$$

2.1.3 Treinamento

2.1.3.1 Feedforward

Na etapa de treinamento de uma rede MLP existem duas etapas principais para que essa seja capaz de aprender a resolver o problema proposto. A primeira etapa é a de *feedforward* onde uma amostra do dado é inserida no modelo, passa por todas as camadas e gera uma saída, que pode ser uma categoria, caso o problema seja de classificação, ou um valor numérico para casos de regressão. Para problemas de classificação não binários, ou seja, com mais de duas classes possíveis como saída, aplica-se a função softmax

$$\text{softmax}(y^t) = \frac{e^{y^t}}{\sum_i e^{y_i}}, \quad (2.4)$$

que aplica uma exponencial normalizada na saída de cada perceptron da última camada e a saída deste modelo é a classe que apresentar o maior valor na saída desta função. A partir dessa saída fornecida pela rede, calcula-se o erro para o valor, ou classe real da amostra inserida. Esse erro é calculado a partir de uma função E e a depender do problema, diferentes funções podem cumprir esse papel. Nos casos de regressão numérica, pode se utilizar a função de erro quadrático médio (MSE), ou $L2Loss$,

$$MSE = \frac{1}{n} \sum_{t=1}^n (y^t - \tilde{y}^t)^2, \quad (2.5)$$

ou a função do erro médio absoluto (MAE), ou $L1Loss$,

$$MAE = \frac{1}{n} \sum_{t=1}^n |y^t - \tilde{y}^t|. \quad (2.6)$$

Já para casos de classificação, utiliza-se a função de entropia cruzada que para casos binários é definida como

$$E^t(\omega|x^t, r^t) = -r^t \log(y^t) - (1 - r^t) \log(1 - y^t), \quad (2.7)$$

onde y^t é uma função de ω e x^t como mostra a Equação 2.1 e r^t é 1 se o exemplo pertence a classe 1, ou seja, $x^t \in C_1$ ou 0 se o exemplo pertence à classe 2, ou $x^t \in C_2$ [19]. Já para casos com mais de duas classes a função de entropia cruzada é definida como

$$E^t((\omega_i)_i|x^t, r^t) = - \sum_i r_i^t \log y_i^t, \quad (2.8)$$

onde neste caso $r_i^t = 1$ se o exemplo pertence à classe i ou seja $x^t \in C_i$ e y é calculado em 2.4 [19].

2.1.3.2 (Backpropagation)

Para otimizar o modelo, emprega-se a técnica da descida do gradiente, caso a função de perda seja diferenciável,

$$\nabla_{\omega} E = \left[\frac{\partial E}{\partial \omega_1}, \frac{\partial E}{\partial \omega_2}, \dots, \frac{\partial E}{\partial \omega_d} \right], \quad (2.9)$$

nela aplicam-se derivadas parciais em relação aos vetores de pesos de cada camada com o objetivo de encontrar os melhores valores para estes pesos de modo a minimizar o erro. Em seguida calcula-se o ajuste desses pesos no modelo

$$\Delta \omega_i = -\eta \nabla_{\omega} E, \forall i \quad (2.10)$$

onde η equivale ao valor da taxa de aprendizagem da rede [19]. E por fim, cada peso ω_i da rede MLP é atualizado

$$\omega_i = \omega_i + \Delta \omega_i. \quad (2.11)$$

Este processo se repete pelo número de épocas definido no treinamento, ou até que a performance do modelo em um determinado conjunto de dados de validação comece a piorar.

Em resumo as Redes Neurais MLP são modelos de aprendizado de máquina que se inspiram no funcionamento do cérebro humano. E assim como tal, cada neurônio artificial recebe informações que são processadas e gera uma resposta. E para aprender a gerar respostas melhores para problemas complexos a MLP aplica a técnica de gradiente descendente para realizar o ajuste fino dos parâmetros de seus "neurônios".

2.2 REDES NEURAS CONVOLUCIONAIS

As *Convolutional Neural Networks* (CNNs) são um tipo de Rede Neural especializada em processamento de dados estruturados em formato de grade, como imagens. As CNNs utilizam Neurônios Convolucionais que realizam operações de Convolução entre a imagem de entrada e filtros ("*Kernels*"), para extrair as características dos dados de entrada, e então, tomar decisões.

Enquanto as redes MLP realizam multiplicações matriciais entre as entradas e as matrizes de pesos, criando uma rede densamente conectada, de maior custo computacional, as convoluções realizadas pelas *ConvNets* "deslizam" o *kernel* através da imagem, filtrando assim, os dados de entrada por partes. A nível de imagem, os pixels são analisados dentro de janela deslizante, e não de uma vez só, como nas redes MLP, que criam um tensor com a dimensão do tamanho dos dados. A Figura 2.3 compara a conectividade entre uma camada densamente conectada versus uma camada convolucional.

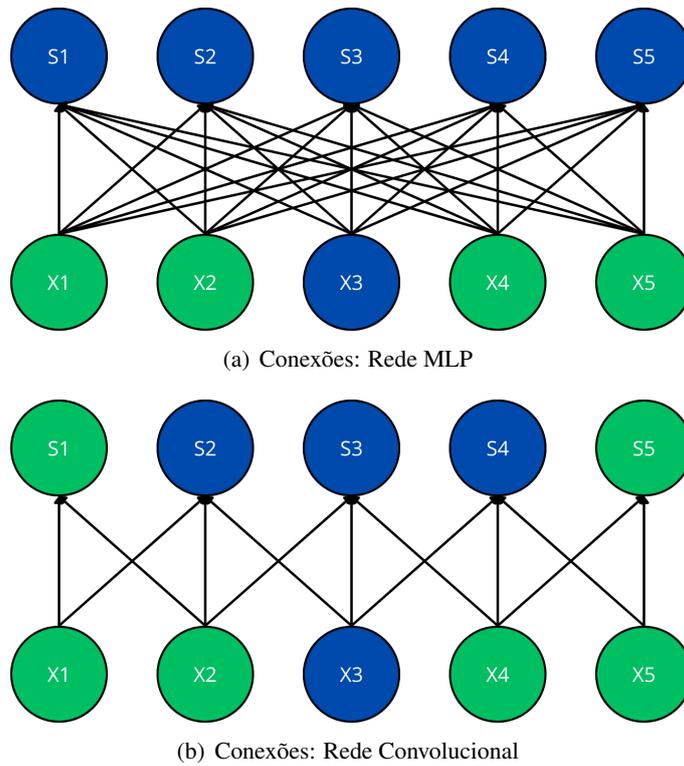


Figura 2.3: Comparativo da densidade de conexões entre redes MLP e redes convolucionais. Em (a), cada neurônio de entrada X_n está diretamente conectado aos neurônios de saída S_n , criando uma rede Totalmente Conectada. Enquanto em (b), os neurônios de entrada X_n estão conectados apenas com alguns neurônios de saída S_n , criando uma rede com conexões esparsas.

Proposta pela primeira vez em 1989 por Yann LeCun, a LeNet [20] foi uma das primeiras Redes Neurais Convolucionais. Criada para reconhecer dígitos numéricos escritos à mão, a LeNet foi treinada em uma extensa base de dados dos Correios Norte-Americano, onde imagem dos dígitos de *zipcode* foram extraídos de correspondências e fornecidos a rede. Apesar de ser uma Rede Convolucional simples para os tempos atuais, a arquitetura da LeNet baseou modelos mais complexos e que são utilizados até hoje. Na Figura 2.4 é possível observar a arquitetura da LeNet. Ela é formada por 7 camadas principais, sendo duas Camadas Convolucionais, duas Camadas de *Pooling*, duas Camadas Densas e uma camada de saída [21].

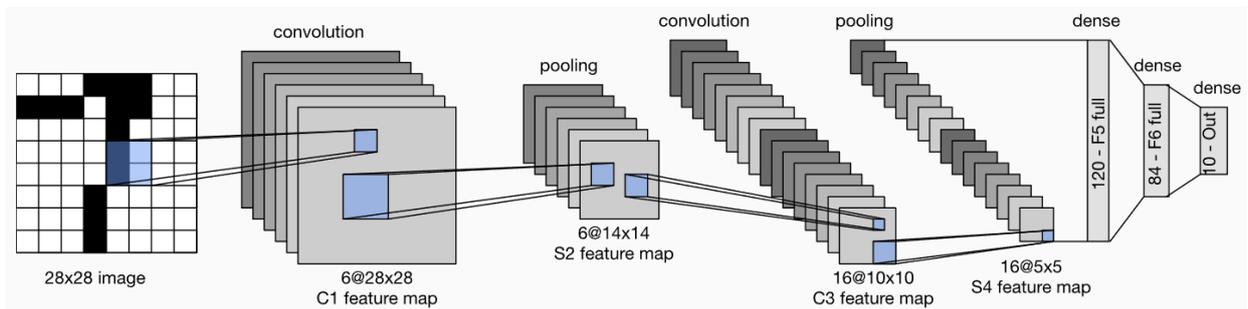


Figura 2.4: Arquitetura da LeNet, a primeira Rede Neural Convolucional implementada. FONTE: ZHANG, Aston & ZACHARY, Lipton. - Dive into Deep Learning

2.2.1 Camadas Convolucionais

Nas Camadas Convolucionais, como é de se esperar, são realizadas operações matemáticas de Convolução. Esta operação correlaciona dois dados diferentes produzindo um terceiro dado formado a partir de multiplicações e somas entre os sinais de entrada.

Normalmente, o processamento realizado por computadores são discretos no tempo. Desta forma, utiliza-se a Convolução Discreta como mecanismo principal desse tipo de camada. Além disso, como as aplicações mais comuns das Redes Neurais Convolucionais estão relacionadas ao processamento de imagens, tanto os dados de entrada quanto os *Kernels* são bi-dimensionais, portanto, a Convolução 2D é frequentemente utilizada para esse tipo de arquitetura.

Analisando a definição de Convolução 2D [22],

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n) \quad (2.12)$$

o sinal $I(m, n)$ seriam as Imagens fornecidas com entrada da rede. O sinal $K(i, j)$ seriam os *Kernels* aplicados à imagem durante a convolução. Por fim, a Saída $S(i, j)$ seria os Mapa de Atributos da Camada Convolutional.

Mesmo utilizando conexões esparsas, os neurônios das camadas mais profundas da rede também estão conectados, de forma indireta, com os neurônios de entrada. Essa conectividade permite que parâmetros obtidos em camadas anteriores se propaguem ao longo da rede, e que, os pesos das camadas finais sofram influência das camadas de entrada. A Figura 2.5 apresenta uma convolução entre dados de entrada (x_n) e um *kernel* de dimensão $K = 3$. No exemplo, é possível observar que, mesmo em camadas distantes, o neurônio g_3 sofre influência dos dados de entrada, pois está conectado de forma indireta à entrada x_n .

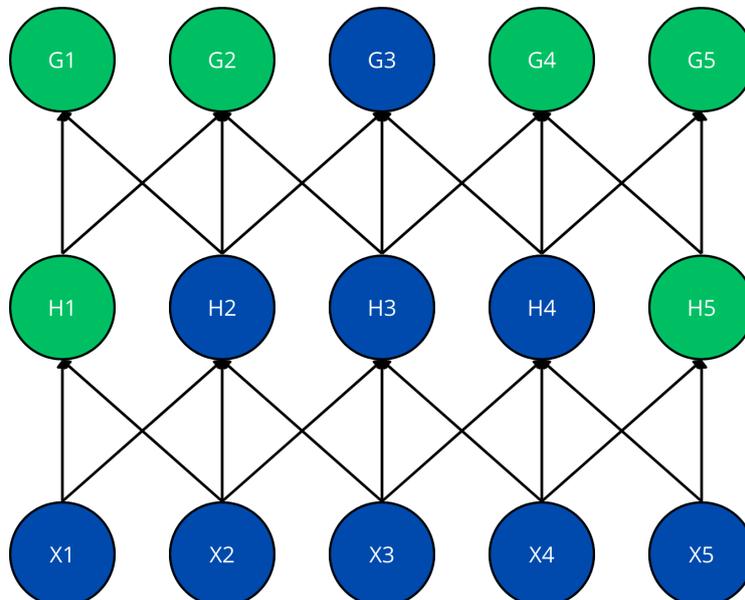


Figura 2.5: Conexão Esparsa: Nem todos os neurônios estão conectados diretamente entre si, porém, mesmo em camadas distantes, o Neurônio g_3 é influenciado pelas entradas x_n , uma vez que estão indiretamente conectados.

Quando um *Kernel* é convoluído com a imagem de entrada, é criado um *feature map* como saída dessa camada convolucional. Cada filtro aplicado possui uma funcionalidade diferente, que permite extrair características específicas da imagem. Numa mesma camada convolucional, é possível aplicar múltiplos *kernels*, gerando na saída múltiplos mapas de atributos, com características diferentes. Durante o treinamento da CNN, os múltiplos filtros aplicados ao longo das camadas convolucionais sofrem ajustes em seus pesos, possibilitando a detecção de características dentro da imagem, como bordas em diferentes direções, texturas, padrões de repetições entre outros tipos de informações interessantes para a tarefa em questão.

Na Figura 2.6 é possível observar um *kernel* 5x5 sendo aplicado em uma imagem 28x28, gerando um mapa de atributos de dimensão 24x24. Neste exemplo, é aplicado apenas um *kernel*, portanto, a dimensão do mapa de atributos na saída dessa camada convolução pode também ser escrita como 1x24x24 [22].

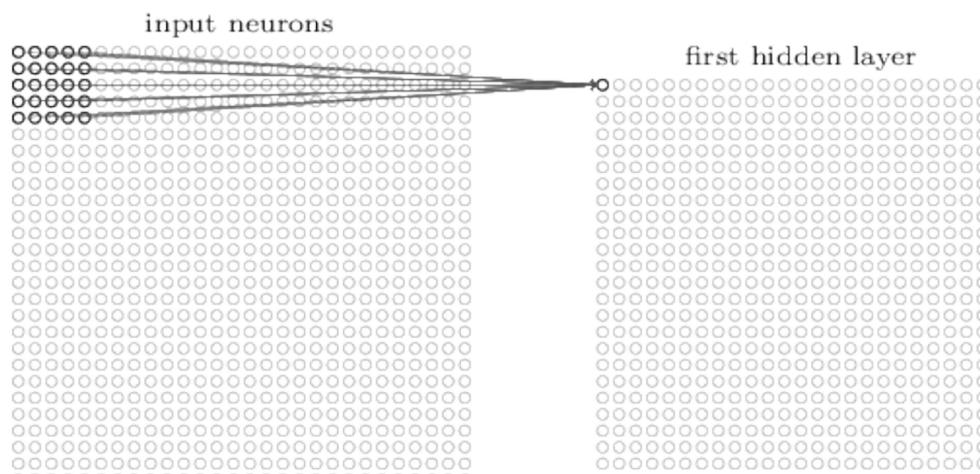


Figura 2.6: Mapa de Atributos: Convolução entre uma entrada 28x28 e Kernel 5x5, criando na saída um mapa de atributos de dimensão 24x24. FONTE: NIELSEN, Michael A. - Neural Networks and Deep Learning.

A LeNet, apresentada anteriormente na Figura 2.4, utiliza 6 *kernels* na primeira camada convolucional (C1) e 16 *kernels* na segunda camada convolucional (C2), porém, CNNs mais modernas utilizam dezenas ou até centenas de filtros por camada convolucional. Arquiteturas de Redes Convolucionais Modernas, como a AlexNet [23] e a VGG16 [24], utilizam quantidades muito maiores de convolução por camada. No caso da AlexNet, em algumas camadas da convolucionais são implementadas 384 convoluções, enquanto nas Redes VGG, o número aumenta para 512 convoluções.

Em resumo, as Camadas Convolucionais dentro de uma Rede Neural Convolucional tem como objetivo principal extrair características da imagem de entrada por meio de convoluções com os Kernels, gerando mapas de atributos que serão utilizados como entrada para as camadas seguintes.

2.2.2 Camadas de Pooling

Pooling são operações muito utilizadas nas Redes Neurais Convolucionais com o intuito de reduzir a dimensionalidade da rede e tornar o modelo mais resistente a invariância. As funções de *pooling* são comumente tratadas como Camadas dentro da arquitetura das CNNs devido a forma como essas Redes são implementadas pelos *frameworks* de Aprendizado de Máquina (PyTorch [25], TensorFlow [26] e Keras

[27]).

Independentemente da nomenclatura utilizada, a Camada de *Pooling*, quando aplicada em algum ponto da rede, substitui as saídas desse camada por novos valores de acordo com a função escolhida. Assim como os filtros utilizados nas camadas convolucionais, é possível controlar a dimensão e *stride* da janela de *pooling*, que deslizará sobre o Mapa de Atributos, sendo possível reduzir ou não as dimensões do novo mapa gerado após o *pooling*.

Além de reduzir a dimensão da Rede, a Camada de *Pooling* tem como objetivo principal tornar o modelo invariante a translações, ou seja, a rede se torna capaz de detectar um objeto, independentemente da posição que este ocupa dentro da imagem. Na Figura 2.7, é possível observar que, apesar do deslocamento dos dados de entrada para direita, a saída permanece igual, sofrendo o mesmo deslocamento que a entrada.

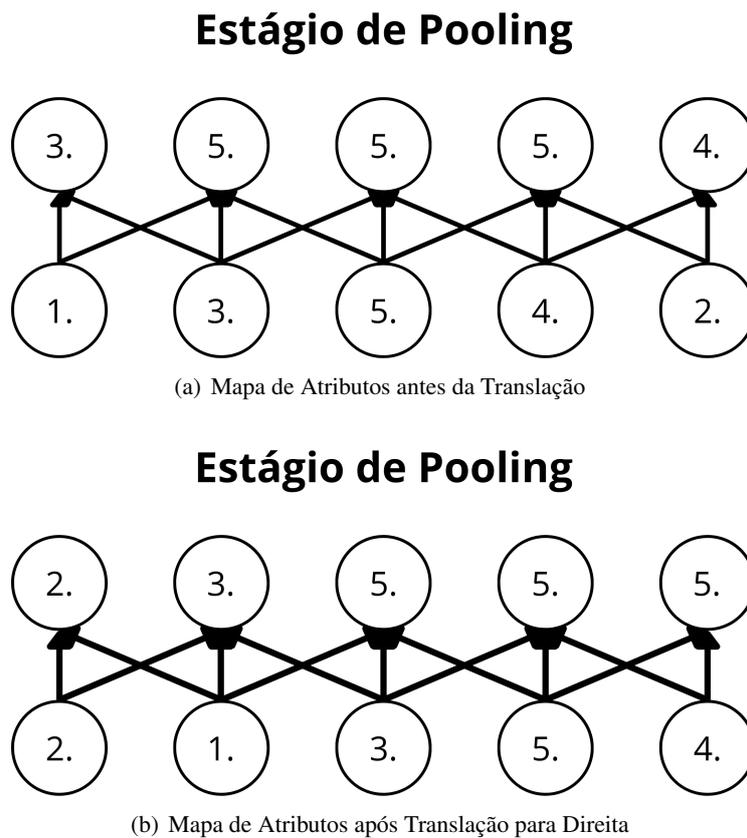


Figura 2.7: *Pooling (MaxPooling)* aplicado em um Mapa de Atributos. A fileira inferior representa a saída da Camada Convolutiva, enquanto a fileira superior representa a saída da Camada de *Pooling*. Na Figura (a) é possível observar que o valor do pixel após Camada de *Pooling* representa o valor máximo entre os pixels central e seus vizinhos. Na Figura (b), o Mapa de Atributos é transladado para direita. Mesmo após o deslocamento, é possível observar que a saída da camada de *pooling* é igual a vista em (a), porém, transladada para direita.

A *Max Pooling*, apesar de simples, é uma das funções de *Pooling* mais utilizadas nas ConvNets. Esta operação permite reduzir as dimensões dos Mapas de Atributos das Camadas Convolucionais, selecionando o valor máximo dentro de uma sub-região desse Mapa, produzindo na saída, um Mapa de Atributos Reduzido. Pode-se observar, na Figura 2.8, um exemplo do *Max Pooling* aplicado a um Mapa de Atributos.

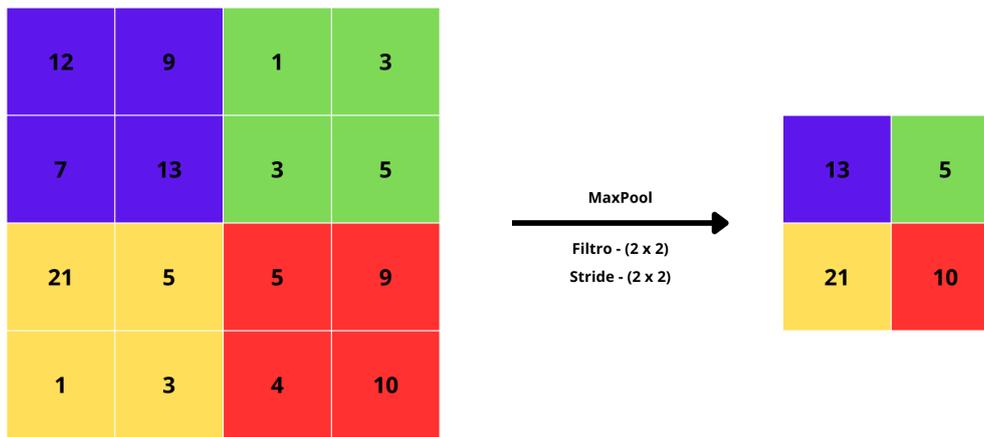


Figura 2.8: Max Pooling aplicado à uma Mapa de Atributos 4x4

As Camadas de *Pooling* assumiram um papel crucial dentro das Redes Neurais Convolucionais. Essas camadas permitem que a rede lide de maneira eficaz com translações e deslocamentos de objetos dentro das imagens, ao mesmo tempo que reduz a dimensões dos mapas de atributos obtidos em camadas anteriores, otimizando ainda mais o processamento da rede nas camadas subsequentes.

2.2.3 Camadas Densas

Assim como visto na Figura 2.4, a arquitetura das CNNs permite combinar as Camadas Convolucionais com Camadas Densas, tradicionalmente utilizadas nas Redes Perceptron de Múltiplas Camadas (MLP), para criar um modelo de Rede Convolucional capaz de extrair características de imagens e então, tomar decisões. As Camadas Densas, ou *Fully Connected Layers*, ao contrário das Camadas Convolucionais, são densamente conectadas, ou seja, cada neurônio é conectado com todos os neurônios da próxima camada oculta. Por este motivo, as Camadas Densas são normalmente adicionadas ao final da rede, uma vez que as operações de Convolução e *Pooling* aplicadas nos Mapas de Atributos ao longo das camadas anteriores, reduziu drasticamente as dimensões dos dados, tornando viável aumentar o número de conexões.

O objetivo final de uma CNN normalmente é a classificação de objetos ou regiões de interesse dentro de uma imagem. Enquanto as camadas convolucionais são especializadas em extrair características e detectar padrões dentro das imagens por meio de filtros aplicados, as camadas densas são utilizadas para classificação das saídas. Nesse sentido, a primeira camada densa da rede recebe os múltiplos mapas de atributos gerados nas camadas convolucionais anteriores. Para que as camadas densas possam processar os mapas bi-dimensionais, é necessário antes vetorizá-los por meio da função de *Flattening*, que realiza a transformação das matrizes de atributos para um vetor unidimensional de tamanho igual a soma do tamanho de todas as matrizes de atributos.

Após a função de *Flattening*, o funcionamento das camadas densas é o mesmo das redes MLP tradicionais. Ela utiliza o vetor unidimensional para criar uma rede MLP densamente conectada. Ainda utilizando a rede LeNet como base, na Figura 2.4, é possível observar que a dimensão do vetor "*flat*" é o mesmo da saída da camada S4, sendo essa dimensão 1x400. Este vetor é passado como entrada da primeira camada densa, composta por 120 neurônios. Nas redes MLP, é comum que a quantidade de neurônios decaia ao

longo das camadas, convergindo para o número de classes que serão modeladas.

Enquanto as camadas convolucionais da rede são utilizadas para extrair características das imagens, as camadas densamente conectadas realizam a tarefa de classificação. As camadas densas recebem como entrada os Mapas de Atributos com padrões e características importantes para a imagem, e funcionam como um classificador final, onde todas as informações extraídas pelas camadas convolucionais são combinadas e transformadas em um vetor de características mais abstratas. As camadas densas são responsáveis por aprender a associar as características extraídas com as classes de interesse, tornando possível a predição da categoria ou classe à qual novas imagem pertencem.

2.3 APRENDIZADO POR MÚLTIPLAS INSTÂNCIAS

O Aprendizado por Múltiplas Instâncias ou *Multiple Instance Learning - MIL*, é um modelo de Aprendizado Fracamente Supervisionado, no qual a rede é treinada com dados organizados em formato de *bags* rotulados. Esse modelo de Aprendizado de Máquina é amplamente utilizado em problemas onde é necessário classificar uma amostra maior com base em padrões que aparecem em apenas algumas sub-regiões, não rotuladas, dessa amostra. Em aplicações médicas por exemplo, podemos definir um Exame de Imagem como o *bag*, enquanto as sub-regiões que formam esse exame, apesar de não possuírem um rótulo individual, são utilizadas para diagnosticar um paciente.

Nos modelos de MIL, o *bag* é formado por múltiplas instâncias menores e sem rótulos, que podem ser definidas como sub-regiões desse *bag*. O objetivo deste tipo rede é classificar um *bag* de instâncias como um todo, e não classificar as instâncias individualmente. A lógica de classificação utilizada por este modelo é simples, no qual um *bag* rotulado como "positivo" é formado por pelo menos uma instância positiva, enquanto um *bag* rotulado como "negativo" é formado por apenas instâncias negativas. Esta lógica de classificação pode ser descrita matematicamente como:

$$Y = \begin{cases} 0, & \text{se } \sum_k y_k = 0 \\ 1, & \text{caso o contrário} \end{cases} \quad (2.13)$$

onde Y é o Rótulo dado ao *bag*. Para que Y seja igual a zero, todas instâncias dentro do *bag* devem ser nulas, para que o somatório mantenha-se em zero. Caso alguma instância tenha valor diferente de zero, a classe escolhida pelo modelo deve ser "1".

Esse tipo de algoritmo de Aprendizado de Máquina é muito utilizado na medicina para classificação de exames de imagem. Normalmente, imagens de biópsias são capturadas em escalas de Gigapixels, onde apenas uma pequena região dela representa um ponto de interesse. Em um cenário onde é preciso determinar se uma biópsia contém ou não células cancerígenas, é improvável que todas as células que formam a amostra em questão sejam cancerígenas. Dessa maneira, as instâncias menores que constituem a biópsia são analisadas individualmente, de modo que, caso alguma delas contenha células cancerígenas, a biópsia como um todo (*bag*) recebe o rótulo correspondente à classe.

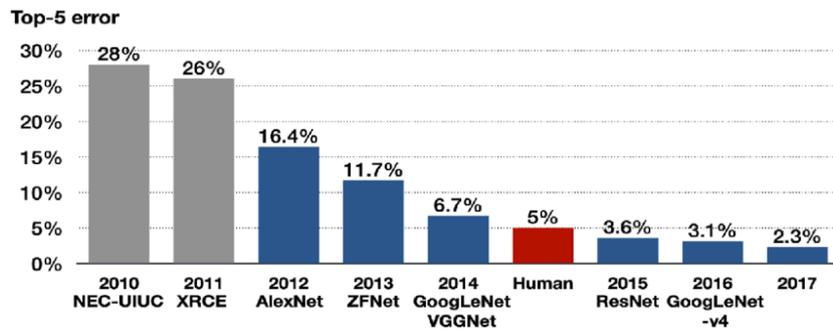
Neste exemplo, na perspectiva de modelo de MIL, o exame de imagem completo representaria o *bag*,

enquanto as sub-regiões menores, ou *tiles*, que formam a imagem inteira, são as instâncias desse bag. Por fim, o diagnóstico da biopsia é utilizada pela rede como Rótulo do *bag*. Durante o treinamento da Rede, são fornecidos como entrada apenas os *tiles*, enquanto na saída deve ser retornada a Classificação do *bag*. Um modelo treinado de Aprendizado por Múltiplas Instâncias deve receber um *bag* contendo instâncias novas e ser capaz de definir um rótulo. Em algumas aplicações médicas específicas, é interessante retornar além das classificações, as sub-regiões marcadas com o rótulo específico, a fim de auxiliar no diagnóstico final.

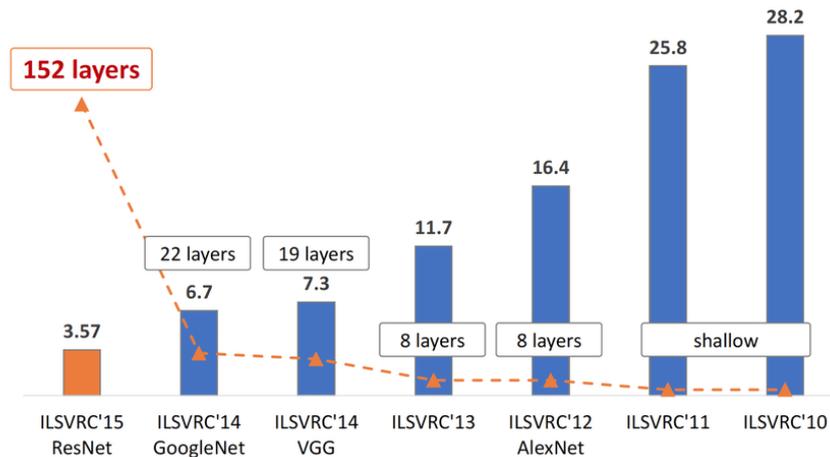
Imagens médicas são capturadas em resoluções altíssimas, garantindo um nível de detalhamento elevado e essencial para serem analisadas por médicos e patologistas. Desta forma, a utilização de *Multiple Instance Learning* em uma Rede Neural Convolutiva possibilita que dados dessa magnitude sejam empregados para treinamento do modelo, uma vez que a imagem é "quebrada" em múltiplas instâncias menores. Além disso, diferentemente das Arquiteturas tradicionais, redes que aplicam o MIL utilizam apenas um rótulo por *bag* ao invés de utilizar rótulos individuais para cada uma das instâncias. Isso facilita que arquiteturas desse tipo sejam ainda mais difundidas entre as aplicações médicas, uma vez que exames de imagem normalmente possuem apenas um rótulo, independentemente de quantas regiões de interesse sejam detectadas.

2.4 REDES RESIDUAIS

Durante muitos anos, utilizou-se a base de dados da ImageNet [28] como desafio para mensurar a performance de algoritmos computacionais na tarefa de classificação de imagens. Essa base de dados é formada por mais de 14 milhões de imagens, divididas em aproximadamente 20 mil classes diferentes. Desde a popularização das CNNs, os resultados obtidos por esse tipo de arquitetura têm sido superior a técnicas anteriores, tornando-se a mais utilizada nesse desafio. Ao longo do tempo, foi possível perceber uma relação entre o aumento no número de camadas convolucionais implementadas pelas redes e a diminuição dos erros de classificação das Arquiteturas vencedoras do ImageNet. A Figura 2.9(a) traz informações sobre o erro de classificação dos modelos utilizados para vencer a competição enquanto a Figura 2.9(b) mostra como a ResNet conseguiu aumentar a profundidade em mais de 6 vezes quando comparada com a VGG e a GoogleNet.



(a) Arquiteturas Vencedoras do desafio do ImageNet ao longo dos anos. FONTE: KANG, Dae-Young & DUONG, Hieu Pham & PARK, Jung-Chul Park - Application of Deep Learning in Dentistry and Implantology



(b) Profundidade das Redes Neurais vencedoras do desafio do ImageNet. FONTE: NGUYEN, Kien & FOOKES, Clinton & ROSS, Aron & SRIDHARAN, Sridha - Iris Recognition with Off-the-Shelf CNN Features: A Deep Learning Perspective

Figura 2.9: Na Figura (a), é possível observar que, a partir de 2012, com a AlexNet, as Redes Convolucionais se sagraram campeãs do desafio. Além disso, nota-se que a ResNet obteve uma acurácia superior a classificação realizada por seres humanos. A Figura (b) mostra a quantidade de camadas utilizadas nos modelos vencedores em cada ano.

Apesar de parecer a solução óbvia, adicionar mais e mais camadas as CNNs não é algo trivial. Conforme observado na Equação 2.9, o cálculo do gradiente depende de múltiplas derivadas parciais, portanto, aumentar deliberadamente a profundidade da Rede pode fazer com que o gradiente assuma valores cada vez menores (ou maiores) ao longo da retro propagação, ocasionando o problema de *vanishing and exploding gradients*, fenômeno nos quais os valores do gradiente são próximos de zero ou assumem valores extremamente altos, resultando em uma atualização mínima ou exagerada dos pesos nas camadas anteriores e degradando o treinamento da rede [29].

As Redes Residuais ou ResNets são arquiteturas moderna das Redes Neurais Convolucionais Profundas que superaram os problemas de degradação causados pelo aumento da profundidade das Redes. Esse modelo utiliza Camadas Residuais que possuem Conexões de Atalho, ou Conexões Residuais, entre a entrada e a saída de um bloco residual, facilitando o treinamento de redes neurais mais profundas.

Redes Convolucionais mais robustas normalmente são segmentadas em Blocos menores. Um Bloco

Residual pode ser representado matematicamente como $H(x) = F(x) + x$, onde $H(x)$ é a saída da Camada, $F(x)$ é a função a ser aprendida pela rede e x a entrada original. A presença do termo x na equação é o que caracteriza a natureza residual da rede. Esse termo aparece na equação pelo fato das Conexões de Atalho ligarem a entrada à saída do bloco. Na Figura 2.10 pode-se observar que a saída do Bloco Residual sofre influência tanto da função $F(x)$ quanto da própria entrada x .

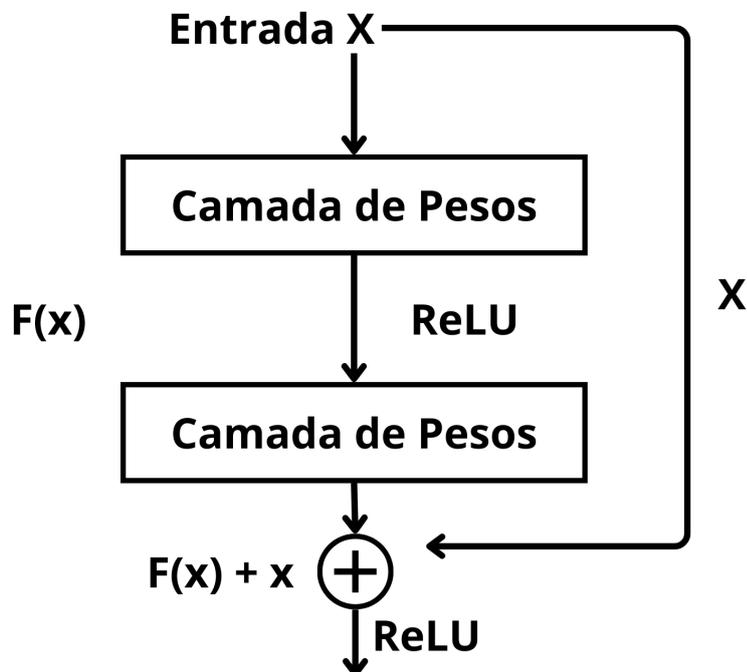


Figura 2.10: Bloco Residual de uma ResNet.

Uma das suposições apresentadas em [29] é que, quando não se conhece a função $H(x)$, treinar um modelo ou um bloco menor para aprender essa função de forma direta pode ser uma tarefa complexa. Reestruturar a arquitetura e adicionar Resíduos ao longo da rede, para que a função $H(x)$ seja igual a $F(x) + x$, torna o aprendizado mais fácil, uma vez que a função $F(x)$ é na maioria das vezes mais simples que $H(x)$. Além disso, a utilização de *Shortcuts Connections* ao longo dos Blocos Residuais servem também para encurtar o caminho feito pelo gradiente na etapa de *backpropagation*, uma vez que esses atalhos reduzem a dimensão da rede em alguns pontos. Reduzir o número de saltos realizados durante a etapa de ajuste dos pesos evita que o gradiente decaia/cresça exageradamente ao longo da rede e assumam valores nulos ou extremos.

De modo geral, a implementação de Camadas Residuais dentro das ResNets permite que redes cada vez mais profundas sejam construídas, sem que ocorra a degradação do gradiente, comprometendo o treinamento da rede. Além disso, a implementação de Redes Neurais mais profundas mostrou-se eficiente em problemas de classificação de imagens, superando até mesmo a performance realizada por seres-humanos.

2.5 MÉTODOS DE AVALIAÇÃO

Nesta seção serão apresentadas as principais métricas e ferramentas de avaliação utilizadas para validar a qualidade de modelos de aprendizado de máquina. A abordagem compreenderá não apenas a forma que cada métrica é calculada, mas também a sua interpretação específica no contexto do modelo em questão, a relevância de cada uma e a justificativa para a sua aplicação.

2.5.1 Matriz de Confusão

A matriz de confusão é uma ferramenta valiosa para a compreensão das métricas que serão contempladas a seguir. Nela, as quantidades de exemplos são dispostas em relação às suas classes reais e às classes previstas. Em contextos de classificação binária, onde apenas duas classes reais estão presentes, a matriz de confusão compreende quatro categorias possíveis, duas em que o valor previsto condiz com o valor real e duas onde esses valores discordam.

Na Figura 2.11, as linhas representam os valores reais, enquanto as colunas representam os valores previstos. Na primeira linha, onde o valor real é negativo, a primeira coluna exibe a quantidade de exemplos dessa classe que foram corretamente classificados como negativos, denominados verdadeiros negativos (VN), ou seja, os verdadeiros negativos (VN). Na segunda coluna desta linha, encontra-se a quantidade de exemplos erroneamente classificados como positivos, caracterizados como falsos positivos (FP). A soma de $VN + FP$ resulta no total de exemplos da classe negativa.

A lógica semelhante se aplica à segunda linha, onde a classe real é positiva. Assim, os exemplos classificados como negativos (primeira coluna) correspondem aos falsos negativos (FN), enquanto aqueles na segunda coluna são classificados como positivos, representando os verdadeiros positivos (VP). A soma de $VP + FN$ proporciona o total de casos da classe positiva.

Para determinar o total de valores previstos para cada classe, basta realizar a soma dos valores por coluna. Desta forma, o total de previsões negativas é obtido pela soma dos valores da primeira coluna, enquanto o total de previsões positivas é alcançado pela soma dos valores da segunda coluna.

Valor Real	Negativo	VN	FP
	Positivo	FN	VP
		Negativo	Positivo
		Valor Previsto	

Figura 2.11: Matriz de Confusão

2.5.2 Precisão

A Precisão é uma medida que avalia a confiabilidade das previsões do modelo para uma classe específica. Essa métrica é calculada pela divisão da quantidade de exemplos classificados corretamente para uma determinada classe pela soma total das previsões feitas para aquela classe. A Equação 2.14 exemplifica o cálculo para os casos positivos, utilizando a nomenclatura apresentada na sub-seção 2.11.

$$\text{precisão}_{\text{classe positiva}} = \frac{VP}{VP + FP} \quad (2.14)$$

$$\text{precisão}_{\text{classe negativa}} = \frac{VN}{VN + FN} \quad (2.15)$$

2.5.3 Revocação ou *Recall*

A métrica de Revocação, também conhecida como *Recall*, avalia a habilidade do modelo em identificar a totalidade das instâncias de uma classe específica que foram corretamente classificadas, em relação ao total de instâncias pertencentes a essa classe. Seu valor é calculado pela divisão do número total de exemplos classificados corretamente e que pertencem à classe em questão pelo total de exemplos existentes na mesma classe, conforme demonstrado pela Equação 2.17, que exemplifica o cálculo para a classe de casos positivos.

$$\text{recall}_{\text{classe positiva}} = \frac{VP}{VP + FN} \quad (2.16)$$

$$\text{recall}_{\text{classe negativa}} = \frac{VN}{VN + FP} \quad (2.17)$$

2.5.4 F1-Score

O *F1-Score* é uma métrica que harmoniza a Precisão e a Revocação, proporcionando uma visão equilibrada do desempenho do modelo. Sua avaliação é derivada da média harmônica entre essas duas métricas, conforme apresentado na Equação 2.18. Essa abordagem harmoniosa torna o *F1-Score* particularmente valioso em situações em que tanto a identificação correta de instâncias positivas quanto a minimização de falsos positivos são cruciais. A fórmula leva em consideração ambas as dimensões da performance do modelo, proporcionando uma medida única e abrangente da eficácia global. Assim, o *F1-Score* oferece uma avaliação mais completa do modelo, equilibrando a capacidade de prever corretamente instâncias positivas com a prevenção de falsos positivos.

$$\text{f1-score} = 2 \times \frac{\text{precisão} \times \text{recall}}{\text{precisão} + \text{recall}} \quad (2.18)$$

2.5.5 Curva ROC e Área Sob a Curva (AUC)

A curva ROC, ou curva de característica de operação do receptor, é um gráfico que mostra o desempenho do modelo avaliado em diversos limiares de decisão por dois parâmetros, a taxa de verdadeiro positivo (TPR), calculada pela equação de *recall* aplicada à classe positiva, definida na Equação 2.17, e a taxa de falso positivo (FPR), que é definida pela divisão da quantidade de exemplos classificados como falso positivo (*FP*) pelo total de exemplos negativos $VN + FP$,

$$\text{FPR} = \frac{FP}{VN + FP}. \quad (2.19)$$

A curva ROC é importante para definir limiares para a classificação e validar quanto o aumento de classificações positivas está impactando na quantidade de falsos positivos e verdadeiros positivos. Uma curva ROC ideal se aproxima rapidamente do canto superior esquerdo do gráfico, indicando uma alta taxa de verdadeiros positivos e uma baixa taxa de falsos positivos. Cada ponto ao longo da curva representa um limiar de decisão específico, permitindo a avaliação da sensibilidade do modelo em relação à especificidade em diversos cenários [30].

Já a área sob a curva (AUC) é uma métrica numérica associada à Curva ROC, refletindo a capacidade global do modelo de classificação. A AUC calcula toda a área abaixo da curva ROC, assim como uma integral do ponto (0,0) ao (1,1) do gráfico, como demonstra a área cinza da Figura 2.12 . Um modelo perfeito, que acerta 100% de suas previsões, terá uma AUC de 1 [30], enquanto um modelo que realiza previsões aleatórias terá uma AUC em torno de 0,5.

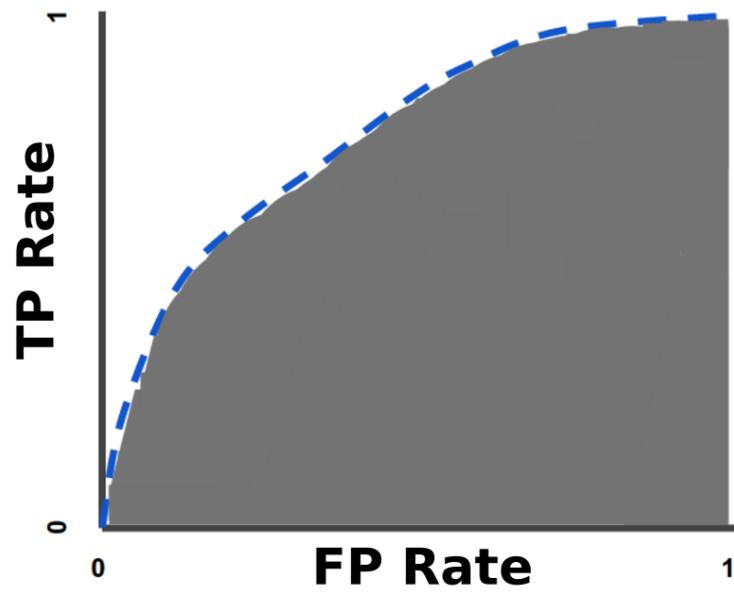


Figura 2.12: Área sob a curva ROC (AUC), FONTE: developers.google.com

3 METODOLOGIA

Neste capítulo serão descritas todas etapas realizadas durante a execução do projeto, desde a obtenção dos dados clínicos até o treinamento da rede. Serão apresentadas as abordagens utilizadas para obtenção dos Exames de Imagem e dos Dados Clínicos, as técnicas de pré-processamento de imagens aplicadas aos Slides obtidos além dos métodos de Descontaminação e Balanceamento desses dados. Serão apresentadas também a Arquitetura da Rede e as estratégias utilizadas durante o Treinamento do modelo.

3.1 OBTENÇÃO DOS DADOS

A qualidade dos dados utilizados durante o treinamento de uma rede neural é um ponto fundamental para sucesso do projeto. Criar uma base de dados do zero seria uma tarefa inviável, pois exigiria uma busca extensa por exames de imagens padronizados, permissões para utilização dos dados dos pacientes e o auxílio de patologistas para validação dos diagnósticos obtidos. Para evitar essa situação, o modelo foi treinado em uma base de dados pública.

O *Genomic Data Commons (GDC)* [31] é um programa de pesquisa do *National Cancer Institute (NCI)* [32]. O GDC têm parceria com diversas instituições médicas, disponibilizando repositórios extensos e confiáveis com Dados Clínicos, Exames de Imagem entre muitos outros tipos de dados médicos úteis para realização de pesquisas. Dentre todos repositórios disponíveis, a base de dados escolhida foi a *The Cancer Genome Atlas - Breast Invasive Carcinoma Collection - TCGA - BRCA*.

O TCGA-BRCA é um repositório específico para pesquisas relacionadas ao câncer de mama e conta com 58.776 arquivos referente a 1.098 pacientes. Dentre os arquivos disponíveis na base de dados do TCGA-BRCA, foram utilizados para o treinamento da Rede 1.978 Slides de Tecido com coloração H&E e 1.098 Arquivos de Dados Clínicos. Cada Slide contém a imagem de uma amostra de tecido cancerígeno disponível em diferentes níveis de resolução, enquanto os arquivos de dados clínicos fornecem o diagnóstico médico e o Status de Receptores de Estrogênio (ER-Status) referente a cada um dos pacientes.

3.1.1 Criando arquivo de Manifesto

O primeiro passo para a aquisição dos dados destinados ao treinamento do modelo é a criação de um arquivo de manifesto. O Portal da *Genomic Data Commons (GDC)* utiliza os arquivos de manifesto, que funcionam como uma "lista de compras" contendo informações dos arquivos que serão baixados no futuro. O manifesto é um arquivo de texto que contém nome, identificadores e informações referente aos arquivos (slides e dados clínicos) em um formato específico para ser utilizado tanto pela *API* (Interface de Programação de Aplicação) quanto pela Ferramenta de Transferência de Dados.

Existem algumas maneiras de se gerar o manifesto referente ao TCGA-BRCA dentro do GDC. Uma delas é acessar o Portal do GDC (portal.gdc.cancer.gov) e clicar na seção "*Projects*". Nela é possível

observar todos os dados disponíveis na base de dados organizados por projetos, sendo possível filtrá-los de acordo com a necessidade. Abaixo estão especificados os passos e filtros utilizados para gerar o manifesto referente ao projeto do TCGA-BRCA.

- **Primary Site:** Foi selecionada a categoria "breast" para apresentar apenas os dados clínicos referente a Mama disponíveis no portal.
- **Program:** Utilizou o filtro do TCGA, para mostrar apenas os dados do TCGA que também estão relacionados ao câncer de mama.
- **TCGA-BRCA:** Aplicando os dois filtros anteriores, já é possível selecionar o projeto TCGA-BRCA.

Após seguir os passos especificados anteriormente, é possível acessar a página do TCGA-BRCA. Na Figura 3.1 pode ser verificado de forma geral, os dados disponibilizados no projeto. Os dados utilizados durante o Treinamento da Rede foram os "Tissue Slide" e "Clinical".

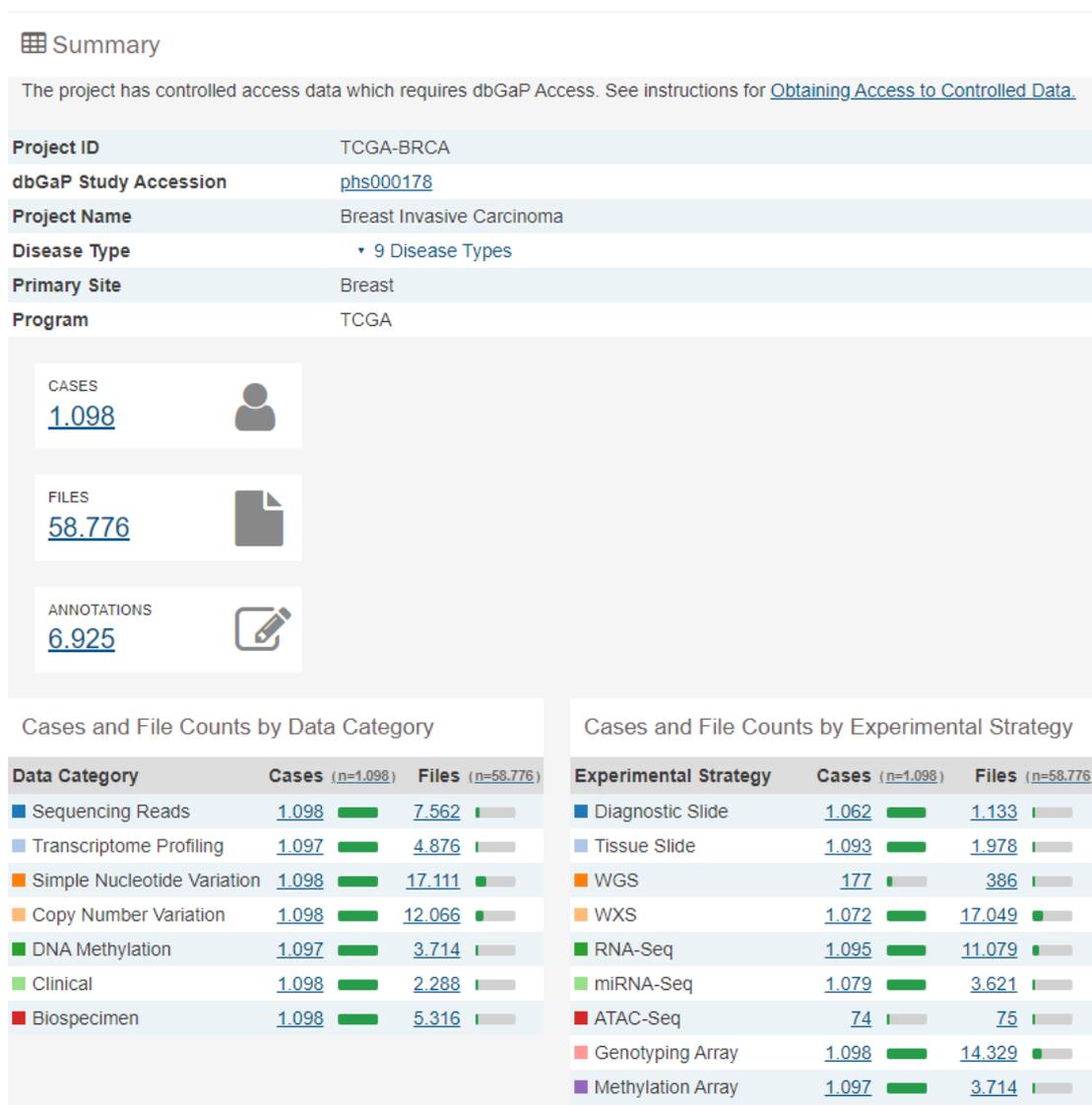


Figura 3.1: Base de Dados do TCGA-BRCA.

Para baixar o manifesto dos Dados Clínicos, é necessário primeiro clicar em "*Clinical*" na página do projeto TCGA-BRCA. Na nova página, serão disponibilizados todos tipos de dados clínicos existentes no projeto. Para filtrar os dados corretos é necessário clicar em "*Files*", no canto superior esquerdo da tela. Logo após é necessário selecionar em "*Data Format*" a opção "*brc xml*". Por fim, basta clicar na opção "*Manifest*", na parte superior direita da tela para iniciar o download. Na Figura 3.2 é possível observar o passo a passo para obter o *Clinical-Manifest*.



Figura 3.2: Base de Dados do TCGA-BRCA.

Para gerar o arquivo de manifesto de Slides, o procedimento é ainda mais simples. A partir da página do TCGA-BRCA, é necessário apenas clicar de "*Tissue Slide*" e em seguida clicar em "*Manifest*", iniciando o download do arquivo de manifesto.

3.1.2 Utilizando o Arquivo de Manifesto

Com base na lista de lâminas selecionadas para treinamento do modelo, conforme detalhado na subseção 3.1.1, a aquisição dos dados foi conduzida de duas maneiras distintas, sendo a primeira delas mais simples enquanto a segunda mostrou-se mais eficiente.

Inicialmente, a primeira abordagem avaliada consistiu no emprego da API do GDC. Nesse contexto, foi implementada uma linha de produção em que os slides seriam baixados e submetidos à transformações apropriadas para viabilizar seu armazenamento de modo a torná-los prontos para a inserção no modelo. A API do GDC representa a interface REST externa do GDC, servindo de base para o funcionamento do GDC Data Portal, do *GDC Submission Portal*, além de estar acessível a usuários externos. Apesar de simplificar o processo de obtenção dos dados, nas fases iniciais do desenvolvimento, a API apresentou uma performance insatisfatória no que concerne à obtenção dos slides.

Por outro lado, o *GDC Data Transfer Tool* se destacou como uma ferramenta otimizada para a aquisição de dados do repositório GDC. Demonstrou um desempenho notável ao efetuar o download das 1.978 lâminas associadas aos 1097 casos resultantes do processo de filtragem detalhado na subseção 3.1.1. Apesar das dificuldades em utilizar a ferramenta, devido a pouca quantidade de manuais ou guias de instalação e utilização disponíveis, uma vez funcionando, a ferramenta de transferência de dados do GDC mostrou-se muito mais eficiente e rápida durante o processo de obtenção dos dados quando comparada com a API.

Devido à performance da Ferramenta de Transferência de Dados em baixar os arquivos presentes nos manifestos, optou-se por essa solução durante o projeto.

3.2 PRÉ-PROCESSAMENTO E TRATAMENTO DOS DADOS

A etapa de pré-processamento é tão importante quanto o treinamento de uma Rede Neural. Garantir a qualidade dos dados e dos diagnósticos, padronizar as amostras, balancear as classes e fragmentar os slides em porções menores é essencial para viabilizar a construção de um modelo capaz de funcionar corretamente. Nas sub-seções a seguir, serão detalhadas todas as etapas de pré-processamento e tratamento dos dados que foram adquiridos no GDC.

3.2.1 Método de Otsu e Remoção de Plano de Fundo

O método de Otsu, ou *Otsu Thresholding* [33], proposto em 1979, é um algoritmo muito utilizado até hoje na área de processamento de imagens para segmentar uma imagem em duas classes distintas: *Background* e *Foreground*. O método encontra de maneira automática um limiar capaz de maximizar a variância inter-classes (ou minimizar a variância intra-classe), de modo que a segmentação entre Objeto e Plano de Fundo possa ser feita verificando apenas os valores dos pixels que formam a imagem.

Esse método transforma a imagem para escala de cinza, de modo que todos os pixels possam ser representados com valores de 0 a 255. Após isso, o algoritmo cria um histograma com a quantidade de vezes que cada intensidade aparece na imagem, mapeando a imagem com base na intensidade dos pixels. Criado o Histograma, o algoritmo seleciona um valor para o limiar T e adiciona os pixels a uma classe com base nesse valor. Pixels com intensidade maior que T pertencem à uma classe e pixels com intensidade menor ou igual a T pertencem a outra. A Variância Intra-Classe é calculada para todos os limiares T possíveis, selecionando aquele que minimiza as Variância Intra-Classes. Minimizar a Variância Intra-Classes, e conseqüentemente maximizar a Variância Inter-Classes, garante que os pixels pertencentes a classe de Plano de Fundo possuam intensidades parecidas, ao mesmo tempo que se diferenciam dos pixels que formam o Objeto.

Ao analisar os slides adquiridos na base de dados do TGCA-BRCA, observou-se que a maior partes das imagens são formada por plano de fundo. A Figura 3.3 mostra um slide obtido nessa base de dados e exemplifica bem essa questão. Além de ocupar mais espaço de armazenamento, áreas de Plano de Fundo não possuem informação, portanto, não contribuem para o treinamento da rede. O Limiar de Otsu foi utilizado para remover o plano de fundo das imagens durante a etapa de *Tiling*, que será abordada em detalhes em seções futuras. Dessa forma, *tiles* da imagem com pelo menos 1% de informação são mantidos e utilizados durante o treinamento da rede, enquanto *tiles* com uma porcentagem de informação menor que 1% foram considerados como Plano de Fundo e descartados.

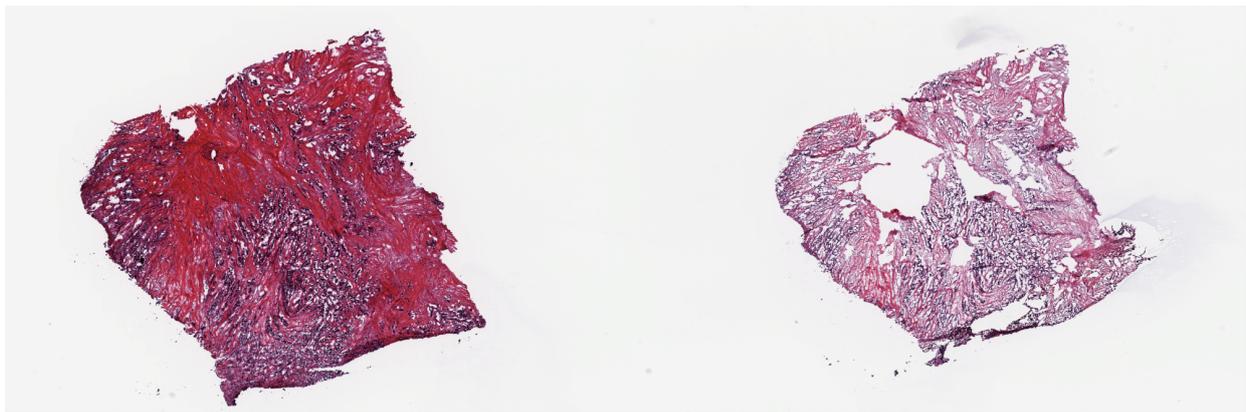


Figura 3.3: Slide de Tecido: A imagem apresenta um slide completo com duas amostras de tecido. O plano de fundo (branco) faz parte do slide original e corresponde a mais da metade da imagem.

Por fim, a Figura 3.4 apresenta 4 *tiles* menores, retirados da Figura 3.3. Nesta imagem, apesar das porcentagens diferentes de informação, todas superam a quantidade mínima de 1%. Além disso, na Figura 3.5 é possível observar os mesmos 4 *tiles* após o método de Otsu. Nelas, os pixels classificados como *background* são tingidos de amarelo, enquanto os pixels classificados como tecido foram tingidos de roxo.

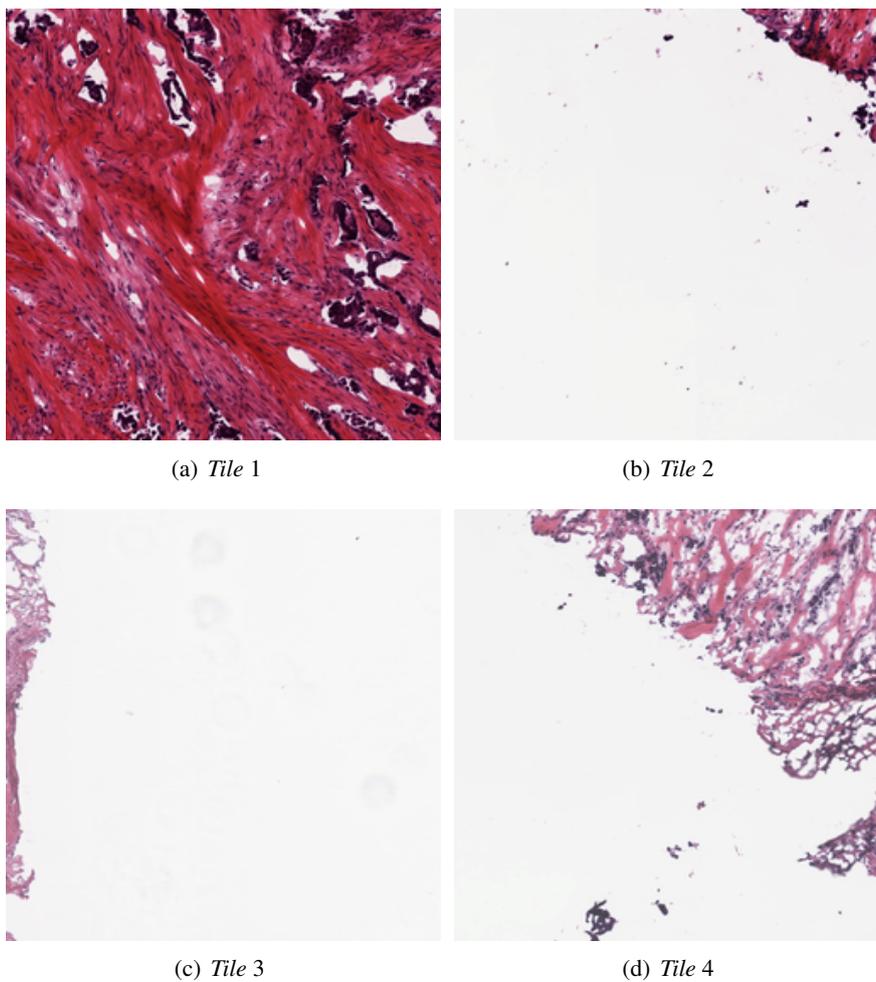


Figura 3.4: Exemplo de quatro *tiles* válidos, gerados a partir de um Slide Inteiro

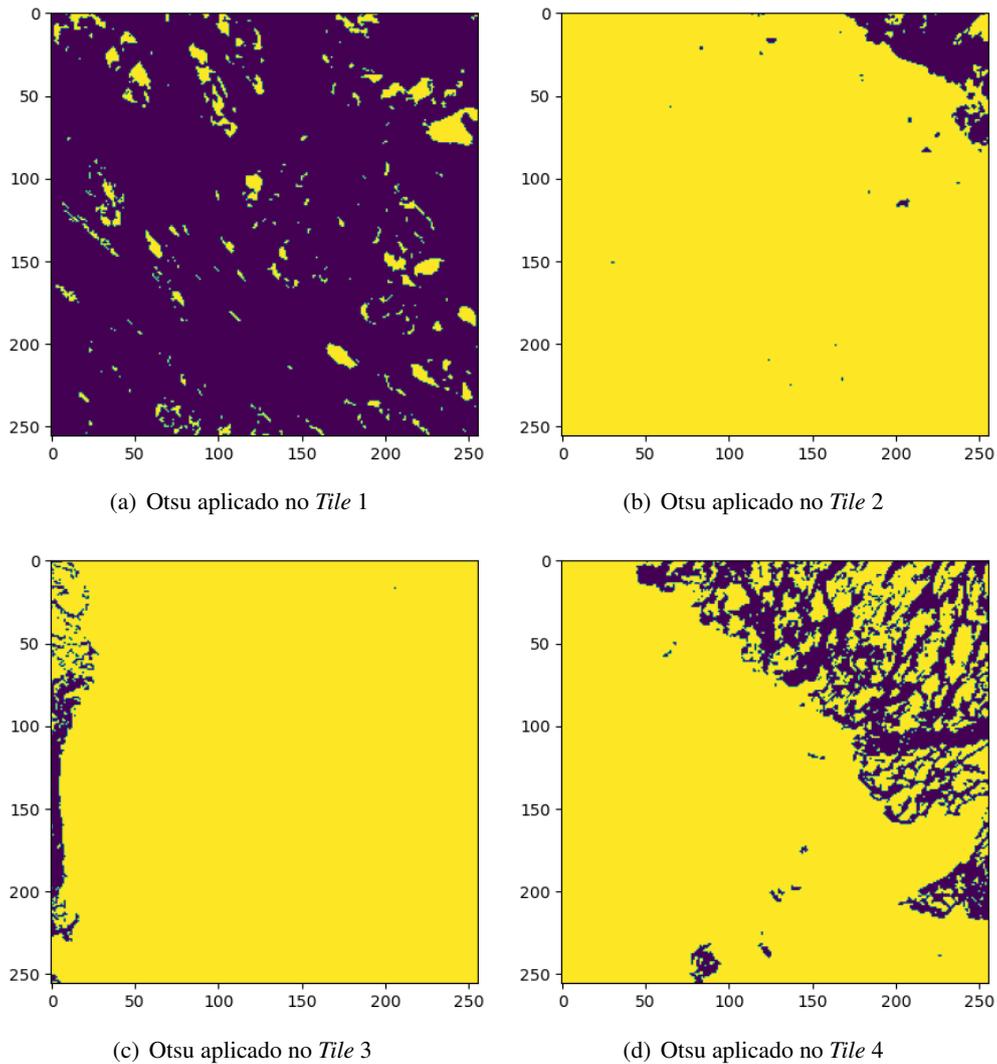


Figura 3.5: Método de Otsu aplicado nos *Tiles*. O Método tingiu de amarelo todos os pixels classificados como background, enquanto os pixels que formam o tecido foram tingidos de roxo.

Utilizar um algoritmo para remoção de plano de fundo garantiu que o treinamento da rede fosse realizado de forma mais eficiente, além de diminuir seu custo computacional e reduzir o espaço de armazenamento reservado para os *Tiles*. Apesar de ser um método antigo, o Otsu ainda é utilizado frequentemente para tarefas de segmentação de imagem, mostrando-se eficiente na detecção de *tiles* de plano de fundo.

3.2.2 Fragmentação/Tiling

A fase de Fragmentação/*Tiling* é a etapa principal durante o pré-processamento dos Slides, dado que é nesta etapa que os *tiles* são gerados. Conforme discutido na seção 2.3, modelos de Aprendizado por Múltiplas Instâncias utilizam *bags* com instâncias menores para realizar o treinamento da rede. Durante a etapa de *Tiling*, são criadas as instâncias que serão utilizadas durante o treinamento e aplicado o método de Otsu (seção 3.2.1), para remoção do plano de fundo.

Devido o fato dos Slides armazenarem imagens médicas em resoluções altíssimas, é inviável utilizá-las

por inteiro durante o treinamento. A Figura 3.3, apresentada anteriormente, é disponibilizada em resoluções que vão de 2558x850 pixels até 81872x27218 pixels. A solução nesse caso foi fragmentar a imagem em "tijolos" com menor resolução e fornecê-los como entrada da rede. Utilizar *tiles* menores além de viabilizar o treinamento da rede com esse tipo de dado, permite que o modelo analise a imagem através de uma lupa, sendo possível a detecção de padrões menores e mais sutis, tornando-o mais preciso

Os Slides foram fragmentados em *tiles* de resolução 256x256 pixels e renomeados com suas coordenadas na imagem original, para facilitar a sua manipulação. Antes de realmente criar cada um dos *tiles*, o método de otsu foi utilizado para calcular o limiar que melhor separa a amostra de tecido e o *background*. Apenas *tiles* com no mínimo 1% de informação foram criados, enquanto *tiles* considerados como plano de fundo foram descartados. Ainda utilizando a Figura 3.3 como exemplo, sem aplicar o *Otsu Thresholding*, seriam criados 34.240 *tiles* de dimensão 256x256, enquanto com o método, foi possível observar o *background* corresponde a aproximadamente 68,6% da imagem, criando-se apenas 10.729 *tiles* contendo informação.

Por fim, para garantir que, tanto a etapa de criação dos *tiles* quanto o método de Otsu estavam funcionando corretamente, foi desenvolvido um algoritmo para reconstruir a imagem do slide utilizando apenas os *tiles* marcados como tecido pelo Otsu. A imagem reconstruída, conforme representada na Figura 3.6, mostra o resultado obtido por meio da utilização do algoritmo. Observa-se que a estrutura dos tecidos permaneceu inalterada, evidenciando a criação correta dos *tiles* que contêm amostras de tecido. Por outro lado, *tiles* que correspondem ao fundo e que não foram criados, foram substituídos por *tiles* pretos.

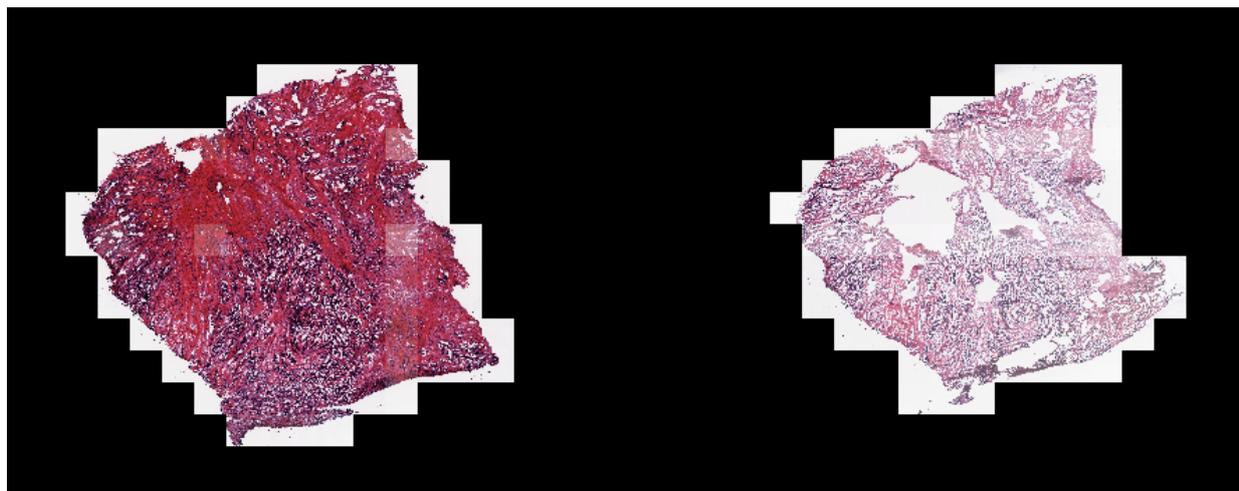


Figura 3.6: Slide Reconstruído: A Figura apresenta imagem reconstruída a partir dos *tiles* com pelo menos 1% de informação. O algoritmo criado substitui os *tiles* inexistentes (*background*) por *tiles* pretos.

3.2.3 Balanceamento dos Dados

Nesta etapa, observou-se uma grande disparidade entre as proporções dos dados, uma vez que 1451 slides apresentaram resultado positivo para os Receptores de Estrogênio, enquanto apenas 439 slides apresentaram um ER-Status Negativo. Isso significa que, para cada slide negativo, existem pelo menos três slides positivos na base de dados do TCGA-BRCA, o que corresponde a uma porcentagem de 76,8% de dados da classe positiva. Treinar um modelo de Rede Neural com um desproporção muito alta entre as

classes pode induzir a um erro de treinamento do modelo, uma vez que durante o treinamento, o modelo pode decidir de forma aleatória pela classe majoritária. Se o treinamento fosse realizado sem nenhum ajuste nas proporções dos dados do TCGA-BRCA, o Modelo poderia aprender a sempre escolher a classe "Positiva", uma vez que atingiria uma acurácia de aproximadamente 76%.

A fim de evitar o treinamento de um preditor aleatório/trivial, utilizou-se a técnica de sobre-amostragem na classe minoritária, que consiste em gerar novos *tiles* a partir da imagem original, de modo que cada *tile* tenha uma área de sobreposição com o seu *tile* adjacente. Essa área de sobreposição possibilita a criação de mais *tiles* para os slides da classe minoritária, que no caso em questão é a classe negativa. Com a utilização desta técnica, aumentou-se a classe negativa em 60% de seu tamanho original, o que não igualou as proporções das classes, mas diminui a diferença de proporções.

3.2.4 Descontaminação dos Dados

Apesar do GDC ser um plataforma renomada que disponibiliza dados médicos de alta qualidade, após a obtenção dos slides e dos dados clínicos, foi realizada uma análise detalhada, confirmando a necessidade de uma limpeza dos dados que seriam usados para treinar o modelo. Conforme citado anteriormente, os dados clínicos fornecem o diagnóstico referente ao ER-Status dos pacientes. Durante a etapa de treinamento da rede, esses diagnósticos são utilizados como Rótulo e, para garantir que a classificação realizada pela rede funcione conforme o esperado, foi necessário filtrar quais tipos de ER-Status seriam considerado válidos.

Dentro da base de dados do TCGA-BRCA, alguns pacientes possuem múltiplos Exames de Imagem, porém, não é disponibilizado nenhum arquivo de Dados Clínicos. Nesses casos, é possível obter as imagens das biópsias desse paciente, mas não há um diagnóstico médico disponível dentro da base de dados. Pacientes com essas características não podem ser utilizados durante o treinamento da rede, uma vez que não é possível fornecer um Rótulo referente aos Slides obtidos.

Além disso, o modelo foi projetado para determinar se há ou não positividade para receptores de estrogênio em pacientes com câncer de mama, de modo que o ER-Status retornado pela rede será "Positivo" ou "Negativo" de acordo com as instâncias fornecidas. Dentre os dados clínicos disponibilizados no portal, alguns pacientes tiveram seus ER-Status marcados como "Empty", "Null" ou "Undetermined". Utilizar esses diagnósticos como Rótulo durante o treinamento da rede adicionaria um grau de complexidade maior ao projeto, uma vez que novas classes seriam modeladas.

Dessa forma, tanto para os casos onde o paciente não possui um arquivo de dados clínicos disponível na base de dados do TCGA-BRCA quanto para dados clínicos onde o ER-Status foram classificados de forma indesejada, optou-se por descartar tanto os Slides quanto os arquivos de Dados Clínicos.

Após tratamento dos dados clínicos, iniciou-se a análise direcionada aos Slides. A descontaminação realizada com foco nos Slides foi mais simples que às etapas realizadas anteriormente, uma vez que, ao eliminarmos os casos de pacientes sem arquivos de diagnósticos ou com ER-Status indesejados, foram excluídos também, os exames de imagem.

Durante a obtenção dos dados, notou-se que um único paciente poderia ter múltiplos exames de imagem. Nesses casos, foi necessário implementar medidas para garantir que, um paciente cujo ao menos um

exame de imagem foi utilizado durante o treinamento da rede, não teriam também, outros Slides utilizados nas etapas de Teste ou Validação da rede. Para isso, realizou-se a divisão dos dados utilizando os ID de Caso dos pacientes e não os slides de forma direta. Caso um paciente seja selecionado para fazer parte do conjunto de dados reservado ao treino, todos os slides desse paciente devem compor o conjunto de treino.

3.3 ARQUITETURA DA REDE

Conforme abordado na seção 1.3, a arquitetura de Rede Neural Convolutiva implementada neste trabalho foi baseada na ReceptorNet. A ReceptorNet é um modelo de Rede Neural Convolutiva Profunda proposta em 2020 [9] para determinar o Status de Receptores de Estrogênio em pacientes com Câncer de Mama utilizando diretamente Exames de Imagem com Coloração H&E. A arquitetura da ReceptorNet pode ser dividida em três redes ou Módulos interconectados, sendo um Módulo de Extração de Características, um Módulo de Atenção e um Módulo de Decisão. A Figura 3.7 apresenta um esquemático geral implementado nesse trabalho, sendo possível observar o processo de criação e limpeza dos *tiles*, além dos módulos que compõem o modelo implementado. Apesar das simplificações realizadas no modelo, a arquitetura implementada manteve-se fiel à proposta no artigo original.

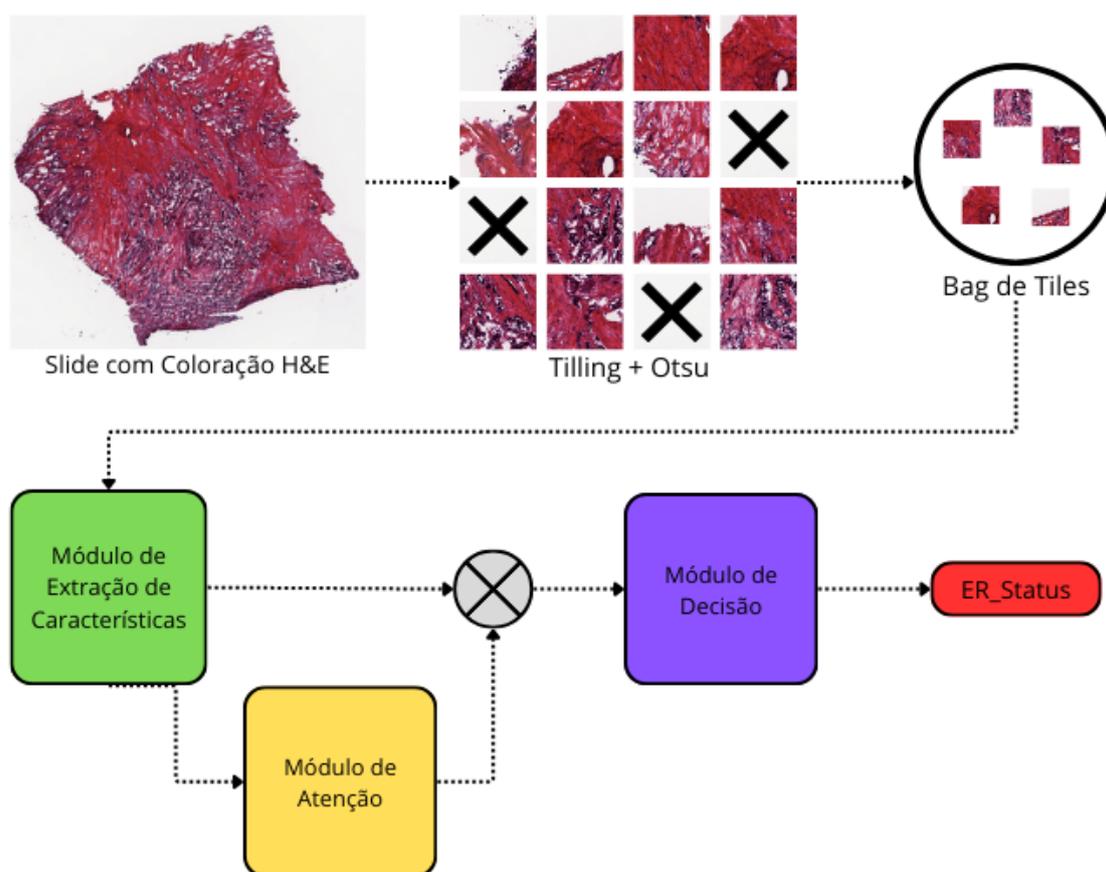


Figura 3.7: Esquemático do Modelo

Como Módulo de Extração de Características da rede, utilizou-se a ResNet-50 em conjunto com duas

camadas densamente conectadas. A ResNet-50 é uma Rede Residual [29] formada por 50 camadas convolucionais, responsáveis por extrair as características das imagens de entrada e fornecer Mapas de Atributos para as camadas seguintes. Apesar da complexidade de implementação de uma Redes Residuais Profunda, o PyTorch oferece a ResNet-50 como um dos modelos disponíveis na biblioteca, sendo possível utilizá-lo de maneira mais simples como extrator de características. Durante a importação da ResNet-50, optou-se por utilizar o modelo pré-treinado na base de dados do ImageNet, proporcionando ao modelo uma inicialização robusta. Além disso, a camada softmax presente na saída da ResNet-50 foi removida e foram adicionadas duas camadas densas que reduzem a dimensão dos tensores que serão fornecidos aos módulos seguintes.

A Figura 3.8 apresenta um esquemático do Módulo de Extração de Características implementado. Um *bag* contendo até 50 *tiles* é passado como entrada para a ResNet-50. Os *tiles* de tamanho 256x256 passam por quatro tipos de blocos residuais, cada um com filtros de tamanhos diferentes, que se repetem ao longo da ResNet-50. Conectadas na saída da ResNet, foram adicionadas duas camadas densamente conectadas, sendo a primeira formada por 1000 perceptrons e segunda formada por 512 perceptrons, que reduzem a dimensão dos tensores para 512.

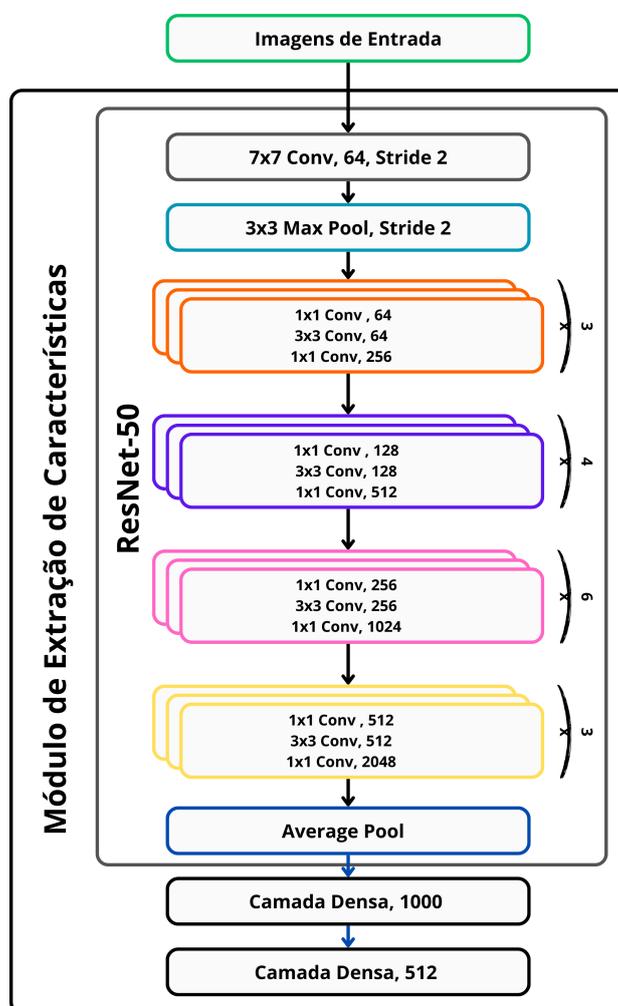


Figura 3.8: Módulo de Extração de Características.

O Módulo de Atenção é adicionado logo após ao Módulo de Extração de Características. Ele recebe como entrada os mapas de atributos gerados no Módulo de Extração e, por meio de algumas camadas densas e uma função Softmax, o Módulo de Atenção cria um Vetor de Atenção na saída. O modelo realiza um produto interno entre o Vetor de Atenção e a saída do Módulo de Extração, que serão fornecidos ao Módulo de Decisão. Os Módulos de Atenção são utilizados em modelos de redes neurais para permitir que, durante a classificação de imagens, o modelo direcione a atenção em áreas específicas da imagem. No caso, o Módulo de Atenção foi aplicado para determinar quais *tiles* são mais importantes dentro dos *bags* para a tomada de decisão, priorizando-os durante a etapa de treinamento.

A Figura 3.9 mostra o Módulo de Atenção implementado. A primeira parte do módulo é uma camada densa formada por 128 perceptrons seguida de uma função de ativação Tangente Hiperbólica. Em seguida, foi adicionada uma segunda camada densa com apenas um perceptron seguido de uma função Softmax.

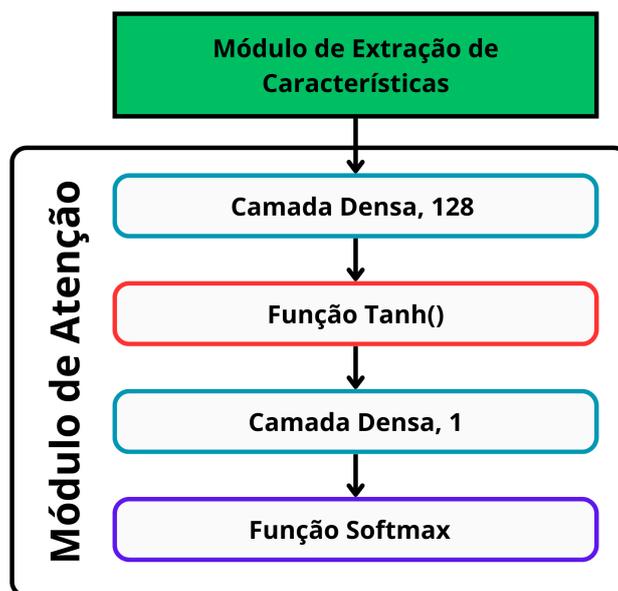


Figura 3.9: Módulo de Atenção

O Módulo de Decisão compõe a porção final do modelo implementado e é responsável por processar o vetor criado a partir do produto interno entre a saída do Módulo de Extração de Características e o Vetor de Atenção gerado no Módulo de Atenção, funcionando como um classificador dos dados de entrada. O Módulo de Decisão é mais simples, sendo implementada uma rede Perceptron Multi-Camadas (MLP) com apenas uma camada de entrada, formada por 512 perceptrons seguida por uma função Sigmoid.

A Figura 3.10 apresenta a arquitetura do Módulo de Decisão implementado. Na Figura é possível observar que o módulo decisão recebe o produto interno entre os tensores criados nos módulos anteriores. Apesar de ser um módulo mais simples, o Módulo de Decisão é responsável por determinar o ER_Status do paciente e executar a tarefa de classificação dentro do modelo.

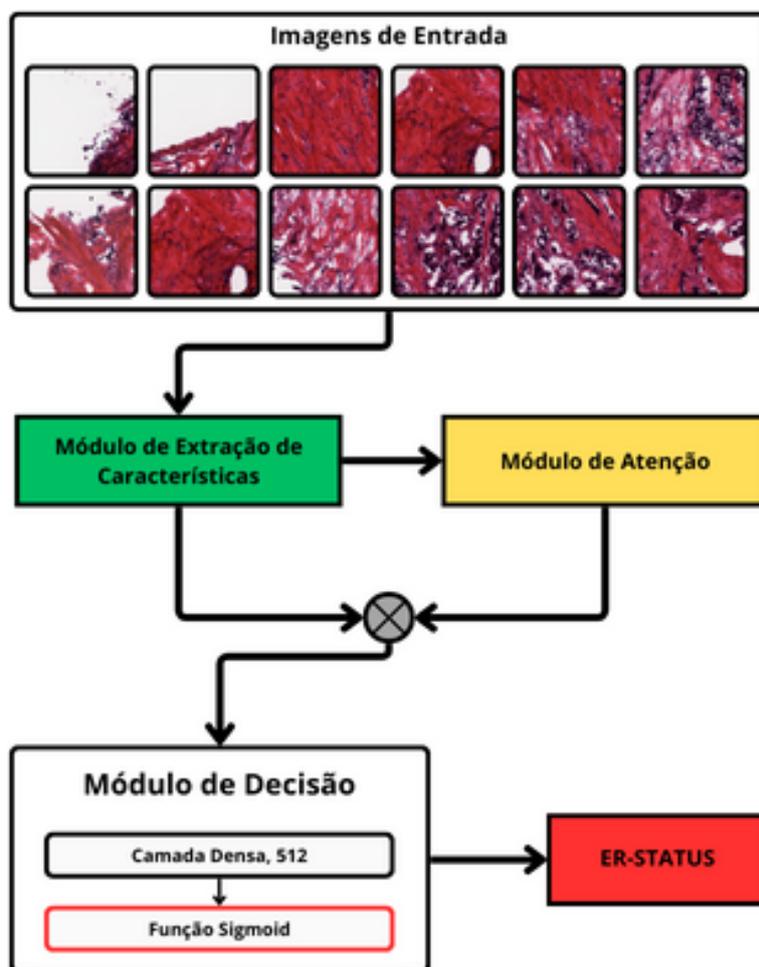


Figura 3.10: Módulo de Decisão

3.4 TREINAMENTO

Finalizada a etapa de Pré-processamento e Tratamento dos Dados conforme seção 3.2 e a implementação dos Módulos de Extração de Características, Módulo de Atenção e Módulo de Decisão conforme seção 3.3, iniciou-se a etapa de Treinamento da Rede.

O treinamento da rede pode ser dividido em três etapas principais, sendo essas: Treinamento, Validação e Teste. Durante a etapa de Treinamento, a rede processa as imagens de entrada e ajusta os pesos das conexões entre os neurônios. A Etapa de Validação é utilizada para avaliar quão bem o modelo generaliza para dados não vistos e para detectar problemas como sobre ajuste ou subajuste. Durante a Validação, os pesos são congelados, ou seja, a rede não sofre nenhum ajuste/aprendizado. Por fim, durante a etapa de Teste, são apresentados um conjunto de dados novo à rede. O conjunto de teste fornece uma avaliação imparcial da capacidade do modelo de generalizar para dados novos e não vistos.

Os *tiles* criados e pré-processados anteriormente foram sub-divididos em 3 conjunto de dados menores, sendo estes Conjunto de Treino, Conjunto Teste e Conjunto Validação, no qual foram separados 80% dos slides pra treinar a rede, enquanto 10% foram separados para Teste e os 10% restantes foram utilizados

para validação do modelo. Na Tabela 3.1 são apresentadas as proporções dos dados separados para cada um dos conjuntos de dados. Os Slides foram distribuídos de forma aleatória entre os três Conjuntos de Dados, respeitando as restrições apresentadas nas seções 3.2.3 e 3.2.4. Apesar do claro desbalanceamento entre as Classes, a utilização de sobre-amostragem em Slides Negativos durante a etapa de balanceamento dos dados aumentou em aproximadamente 60% o número de *tiles* referentes a classe minoritária.

	Quantidade de Slides	Slides ER_Status Positivo	Slides ER_Status Negativo
Conjunto de Treino	1515	1166	349
Conjunto de Validação	184	137	47
Conjunto de Teste	189	147	42
Total	1888	1450	438

Tabela 3.1: Quantidade de Slides em cada Conjunto de Dados.

Para o treinamento da Rede, foi adotado o método de Aprendizado por Múltiplas Instâncias. Desta forma, foram fornecidos como dados de entrada, *bags* contendo até 50 *tiles* aleatórios de dimensão 256x256, retirados de um Slide de Tecido. Se uma amostra de tecido é formada por mais de 50 *tiles* válidos, novos *bags* são criados enquanto houverem *tiles* sobrando.

Durante o treinamento, a rede processa os *bags*, ajustando os pesos das conexões entre os neurônios, a fim de obter um desempenho melhor na tarefa de classificação. Ajustar os pesos da ResNet-50 durante a etapa de treinamento tem custo computacional elevado devido a profundidade e a quantidade de filtros utilizados em cada camada, por isso, optou-se por congelar o gradiente no módulo de extração de características, ou seja, mesmo durante a etapa de treinamento da rede, os pesos se mantiveram constante com os valores importados. Os pesos das conexões dos neurônios nos Módulo de Atenção e Decisão foram ajustados livremente. Como otimizador, foi escolhido o Adam (*Adaptive Moment Estimation*), que é uma extensão do Gradiente Estocástico que ajusta dinamicamente a Taxas de Aprendizado ao longo do treinamento. A Função de Custo, responsável por mensurar o Erro do modelo, foi a Entropia Cruzada, comum em tarefas de classificação. Ela mede a diferença entre as distribuições de probabilidade previstas pelo modelo e as distribuições reais dos rótulos de classe. Ambos otimizador e função de custo foram mantidos iguais aos especificados no artigo.

Os Slides adquiridos podem ser acessados em quatro níveis distintos de resolução. Na Tabela 3.2, são detalhados os níveis disponíveis, juntamente com as resoluções correspondentes para cada um. Embora resoluções mais elevadas proporcionem um treinamento mais eficaz da rede, o aumento na quantidade de *tiles* resulta em um custo computacional mais elevado, tornando o treinamento impraticável em um computador pessoal. O aumento da resolução para um nível acima corresponde a aumentar a quantidade de *tiles* em aproximadamente 7.5 vezes. Por este motivo, optou-se por treinar a rede utilizando slides com resolução de nível 3.

Nível	Resolução (Pixeis)	Total de <i>tiles</i> (256x256)	<i>tiles</i> Úteis
0	87.968 x 35.840	48.108	24.054
1	43.991 x 17.930	12.036	6.018
2	10.997 x 4.482	753	377
3	2.749 x 1120	47	24

Tabela 3.2: Comparação entre os Níveis dos Slides e Resolução correspondente. A Resolução em cada nível varia para cada Slide diferente, porém, os valores apresentados foram obtidos por meio de cálculo de média simples entre 10 Slides aleatórios.

Para comparar o desempenho do modelo simplificado da ReceptorNet desenvolvido neste trabalho, foi implementada uma segunda arquitetura de Rede Neural Convolutiva (Rede de Controle). Os resultados desta segunda rede também serão apresentados na seção de Resultados. A Rede de Controle é composta por um Módulo de Extração (ResNet-50) e um Módulo de Decisão (Rede MLP). Nessa arquitetura, cada *tile* recebeu um rótulo individualmente, seguindo uma abordagem semelhante à de uma rede CNN tradicional, portanto, não foi empregado o Aprendizado por Múltiplas Instâncias.

4 RESULTADOS E DISCUSSÕES

Neste capítulo, serão delineados e discutidos os experimentos conduzidos, juntamente com os respectivos resultados obtidos, proporcionando uma visão das análises realizadas. Cada fase do desenvolvimento do modelo será devidamente abordada, oferecendo uma compreensão de sua concepção, implementação e desempenho. A contextualização das etapas do processo e a análise crítica dos resultados visam comparar os resultados obtidos nesta implementação do modelo obtido com aquele proposto em [9] assim como enriquecer a compreensão do leitor sobre a eficácia do modelo obtido.

4.1 REDE DE CONTROLE

Utilizando a arquitetura da Rede de Controle apresentada na Seção 3.4, o modelo foi treinado durante 30 épocas, onde se percebeu que a função de perda do modelo continuou a abaixar enquanto sua taxa de erro, definida como

$$error = \frac{FPR + FNR}{2}, \quad (4.1)$$

passou a aumentar, como se observa entre as épocas 25 e 30 na Figura 4.1, indicando um ponto de parada. Como resultado, obteve-se um modelo que classificou arbitrariamente a maior parte dos pacientes como positivo devido ao desbalanceamento inicial dos dados.

Ao comparar as classes reais com as classes previstas pelo modelo, identificou-se que, dentre os 189 pacientes que compunham o conjunto de teste, 133 foram acertadamente classificados como positivos, enquanto 14 pacientes que de fato eram positivos foram erroneamente classificados como negativos. Ao analisar os pacientes cujo diagnóstico real era negativo, observou-se que apenas 9 foram classificados corretamente, enquanto 33 foram incorretamente classificados. Essa análise é refletida na Figura 4.2, que exibe uma Matriz de Confusão, proporcionando uma visualização dos Rótulos Originais em comparação com as Predições realizadas pelo modelo.

A partir das previsões exibidas na matriz de confusão é possível calcular as métricas apresentadas na Seção 2.5. Dessa forma, para a precisão foi calculada uma precisão de 80% com *recall* de 90% resultando em um *f1-score* de 85%, contudo para a classe negativa a precisão foi de 39% com *recall* de 21% resultando em um *f1-score* de 28%. Analisando a curva ROC, na Figura 4.3 nota-se que a Área sob a curva (AUC) aponta uma performance de 0,56. Esses resultados comprovam que o modelo está, em grande parte, realizando previsões indiscriminadas para a classe majoritária, no caso a positiva, sem de fato conseguir aprender regras para generalizar ambas as classes de maneira eficaz.

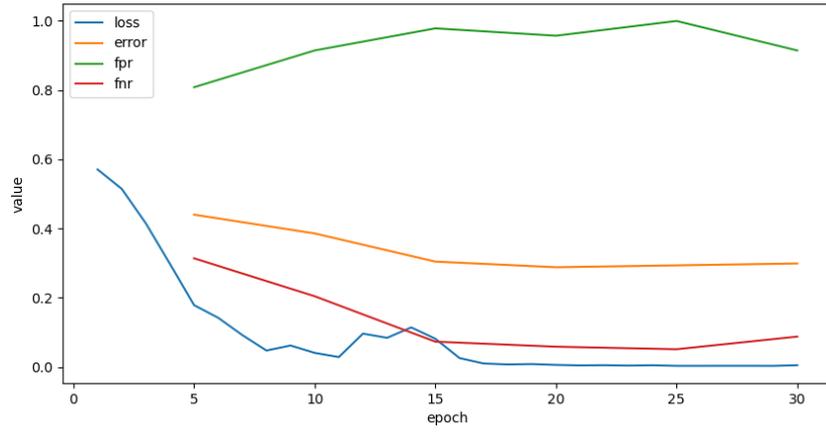


Figura 4.1: Curvas de Custo (*loss*), Erro (*error*), Taxa de Falsos Positivo (FPR) e Taxa de Falsos Negativo (FNR) a cada época de treinamento do modelo de 30 épocas

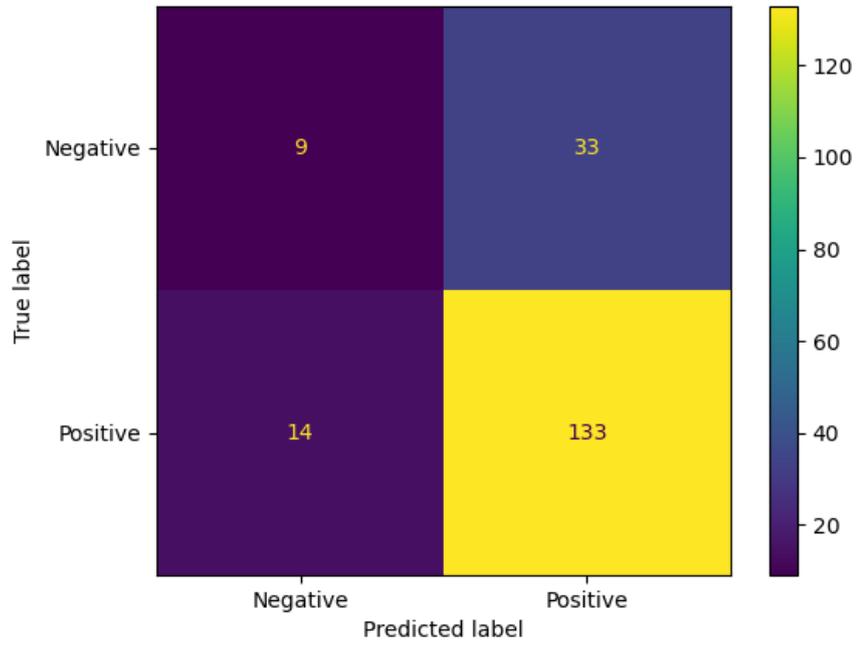


Figura 4.2: Matriz de Confusão da Rede Neural Convolutacional Convencional

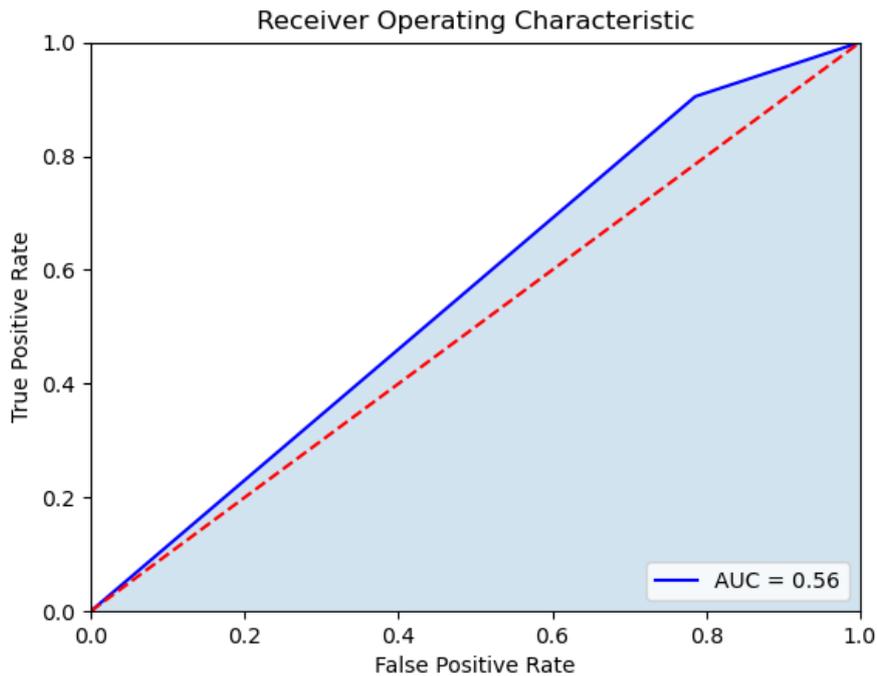


Figura 4.3: Curva ROC

Também foi treinado um modelo utilizando a mesma arquitetura por 105 épocas para validar se o modelo treinado por 30 épocas não havia convergido em um mínimo local da função de otimização. Contudo, os resultados obtidos foram similares como exibe a matriz de confusão na Figura 4.5. As curvas de erro (*loss*) e taxas de falsos positivo (*fpr*) e falsos negativo (*fnr*) como mostra a Figura 4.4 oscilam em torno de uma média mas não chegam a convergir para um mínimo ao longo do treinamento.

Assim como no modelo anterior, foram extraídas métricas a partir da matriz de confusão, dessa forma, para a classe positiva a precisão foi de 77,8% com *recall* de 87,8% e *f1-score* de 82,4% enquanto para a classe negativa a precisão foi de 21,7%, *recall* de 11,9% resultando ao *f1-score* de 15,4%. Comprovando o comportamento do modelo treinado com menos épocas.

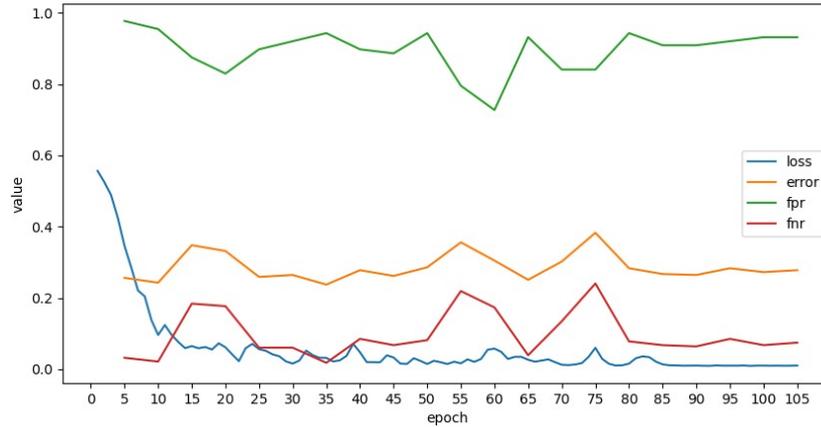


Figura 4.4: Curvas de Custo (loss), Erro (error), Taxa de Falsos Positivo (FPR) e Taxa de Falsos Negativo (FNR) a cada época de treinamento do modelo de 105 épocas

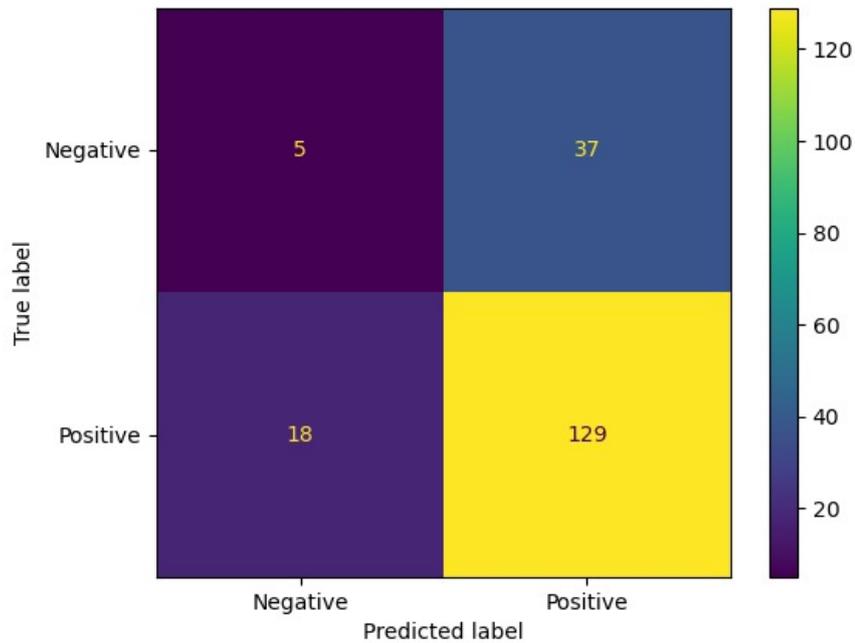


Figura 4.5: Matriz de Confusão da Rede Neural Convolutonal Convencional

4.2 RECEPTORNET SIMPLIFICADA

Após aplicar as métricas de validação no Modelo de Controle, os mesmos testes foram realizados no modelo simplificado da ReceptorNet implementado nesse trabalho. A arquitetura descrita na Seção 3.3 passou por dois cenários de treinamentos diferentes, sendo o primeiro treinamento realizado durante 50 épocas enquanto o segundo treinamento foi realizado por 100 épocas.

A Figura 4.6 apresenta as Curvas obtidas durante o treinamento da Rede durante 50 épocas. Para todas as curvas presente na Figura é possível notar um comportamento muito mais estável quando comparado com o modelo de CNN tradicional.

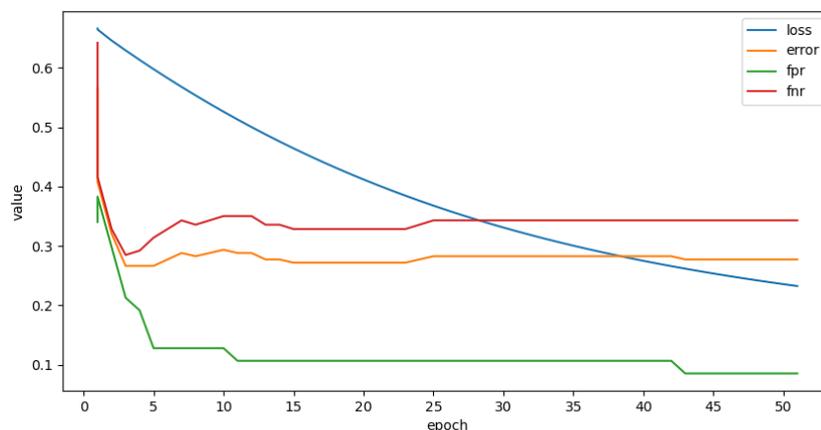


Figura 4.6: Curvas de Custo (loss), Erro (error), Taxa de Falsos Positivo (FPR) e Taxa de Falsos Negativo (FNR) a cada época de treinamento do modelo final treinado por 50 épocas

Aplicando o modelo treinado ao conjunto de teste, 38 amostras foram corretamente classificadas como negativas, enquanto apenas 4 amostras negativas foram classificadas erroneamente como positivas, resultando em uma precisão de 49,3%, *recall* de 90,5% e *f1-score* de 63,9% para a classe negativa. Quanto à classe positiva, 108 amostras foram classificadas corretamente enquanto apenas 39 foram classificadas como falso negativo, resultando em uma precisão de 96,4% com *recall* de 73,5% e *f1-score* de 83,4%. Os cálculos foram realizados com base na matriz de confusão da Figura 4.7.

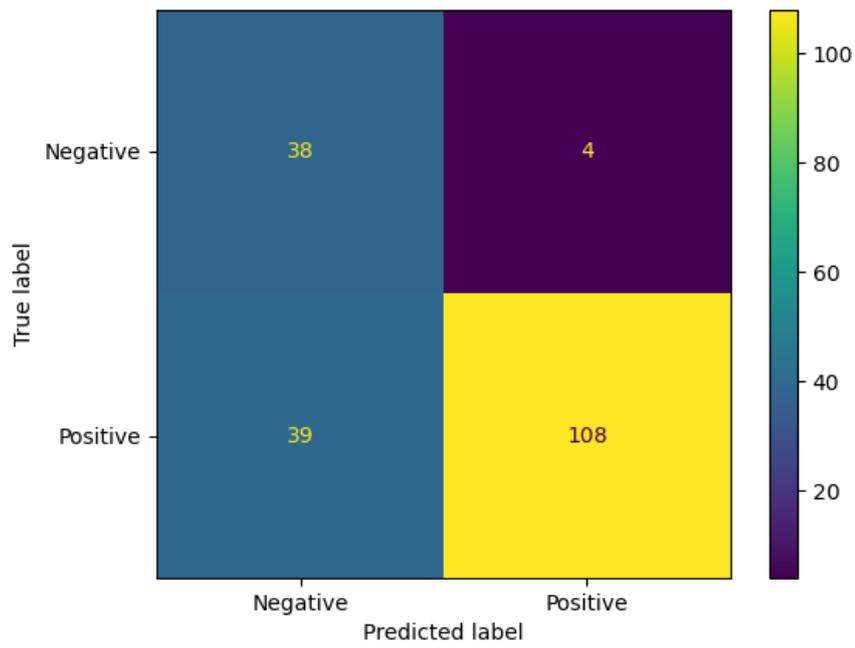


Figura 4.7: Matriz de Confusão da ReceptorNet Simplificada

Observando a curva ROC apresentada na Figura 4.8 a área sob a curva (AUC) foi de 0.82, confirmando o bom desempenho do modelo.

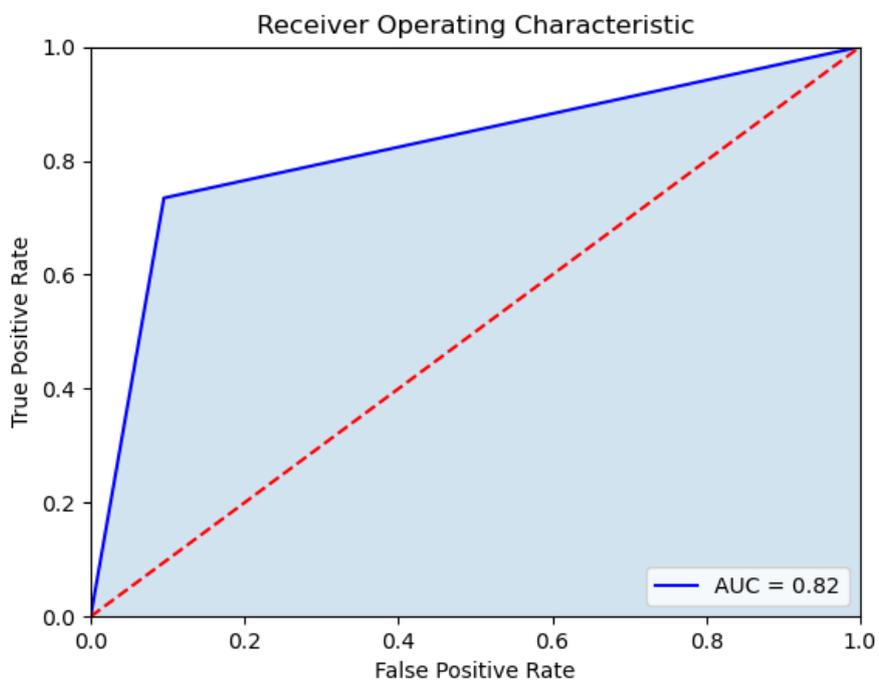


Figura 4.8: Curva ROC

O segundo treinamento desta arquitetura foi feito por 100 épocas. A Figura 4.9 traz as curvas obtidas durante a etapa de validação, onde é possível perceber que, a partir da época 25, o erro tende a aumentar.

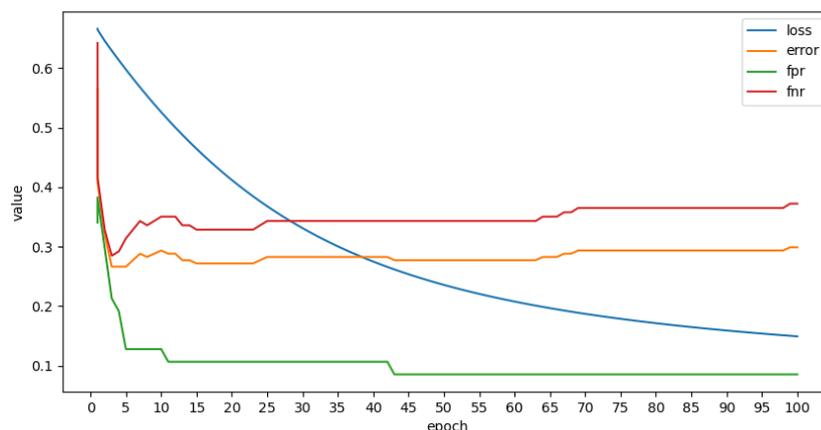


Figura 4.9: Curvas de Perda (*loss*), Erro (*error*), Taxa de Falsos Positivo (FPR) e Taxa de Falsos Negativo (FNR) a cada época de treinamento do modelo final treinado por 100 épocas

Observando as previsões do modelo quanto aos dados de teste, na Figura 4.10 nota-se que há apenas um novo erro na classe positiva, aumentando os falsos negativo para 40 e reduzindo os verdadeiros positivo para 107. Calculando as novas métricas para a classe positiva, obteve-se precisão de 96,4% *recall* de 72,8% e *f1-score* de 82,9%. E analisando a curva ROC na Figura 4.11, observa-se que a área sob a curva (AUC) se manteve inalterada.

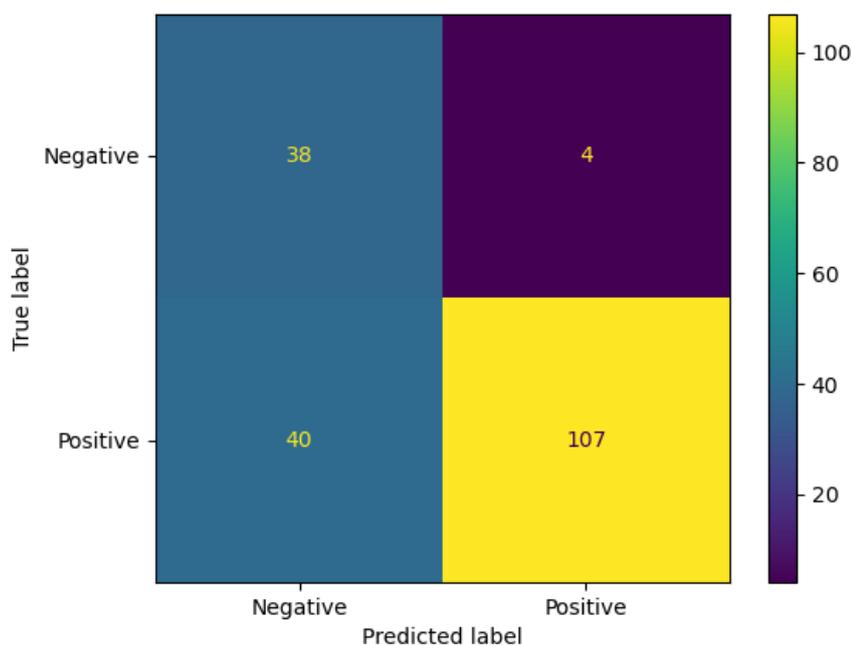


Figura 4.10: Matriz de Confusão da ReceptorNet Simplificada

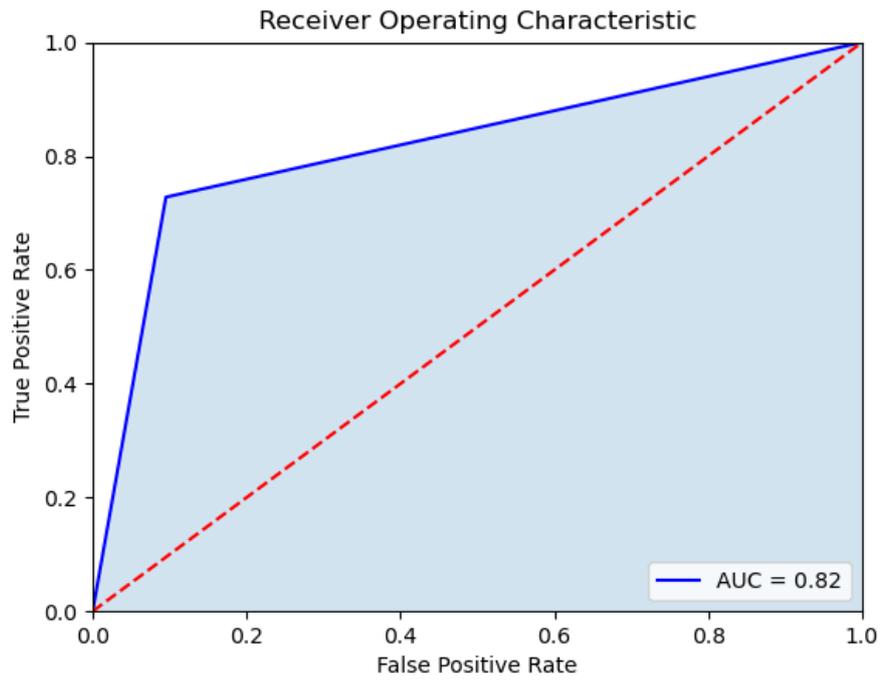


Figura 4.11: Curva ROC

5 CONCLUSÃO E TRABALHOS FUTUROS

O objetivo principal deste trabalho foi criar um modelo de Rede Neural Profunda capaz de determinar o Status de Receptores de Estrogênio em pacientes com Câncer de Mama utilizando diretamente Exames de Imagem com Coloração H&E. Apesar da complexidade em classificar tecidos cancerígenos a partir da morfologia celular, algoritmos com esse objetivo já foram implementados anteriormente e obtiveram taxas de sucesso elevado, como a ReceptorNet [9], que inspirou diretamente o modelo implementado nesse trabalho. Ao longo deste documento, foi apresentada uma solução capaz de realizar esse mesmo tipo de classificação, porém, executando todas etapas de pré-processamento e treinamento da rede em uma máquina com poder computacional limitado, sendo necessário realizar algumas simplificações. Mesmo com as limitações de *hardware*, os resultados obtidos foram satisfatórios.

Implementar este tipo de algoritmo só é viável se houver uma grande quantidade de dados clínicos e exames de imagem de alta qualidade disponíveis. Para isso, foi necessário obter os dados do BRCA-TCGA, disponíveis na base de dados pública do GDC. Apesar da qualidade elevada, os dados adquiridos eram fortemente desbalanceados, sendo necessário realizar diferentes procedimentos de balanceamento e descontaminação. Além do balanceamento e descontaminação dos dados obtidos, para suprir as condições de um modelo de Aprendizado por Múltiplas Instâncias foi necessário quebrar os exames de imagem em fragmentos menores chamados *tiles*. Por fim, além de fragmentar os Slides obtidos, foi necessário remover *tiles* de plano de fundo, uma vez que esses não correspondem à amostras de tecido e não contribuem para o treinamento da rede.

Finalizada a etapa de obtenção e pré-processamento dos dados clínicos e exames de imagens, iniciou-se a implementação da Rede Neural Convolutiva Profunda. Durante as pesquisas realizadas sobre o assunto, foi possível notar a preferência por utilização de modelos de Aprendizado por Múltiplas Instâncias em tarefas de classificação de tecidos cancerígenos, uma vez que este tipo de modelo utiliza instâncias menores de um conjunto maior para realizar o aprendizado e classificação. Se ao menos uma dessas instâncias forem classificadas como positiva, todo conjunto recebe a mesma classificação. Em outras palavras, se tecidos cancerígenos hormônio-positivos forem detectados em uma pequena parcela da imagem analisada, toda a amostra é classificada como positiva para receptores hormonais. Este tipo de classificação se assemelha ao processo de classificação realizado por patologistas.

A Rede Neural implementada neste trabalho pode ser sub-dividida em três módulos distintos, cada um com funcionalidades específicas, sendo eles um Módulo de Extração de Características, um Módulo de Atenção e um Módulo de Decisão. O módulo de Extração de Características recebe um *bag* contendo alguns *tiles* da Imagem de entrada. Para o módulo de Extração foi utilizada a ResNet-50, uma Rede Residual profunda capaz de detectar e extrair padrões e características das imagens fornecidas. O Módulo de Atenção foi utilizado para direcionar a atenção da rede para determinados *tiles* dentro do *bag* durante a etapa de treinamento. Esse módulo é conectado à saída do Módulo de Extração de Características e serve para ajudar a detectar quais *tiles* são mais importantes para tarefa de classificação. Por fim, o Módulo de Decisão forma a porção final da arquitetura e é responsável pela classificação do *bag* processado. Ele recebe todos os tensores processados nos módulos anteriores e, por meio de uma rede MLP, gera na saída

o Status de Receptores de Estrogênio referente ao paciente.

Após concluir a fase de treinamento da rede, foi possível analisar as métricas e resultados obtidos. Apesar das simplificações realizadas no modelo em relação à proposta original do artigo da Nature [9], o modelo implementado demonstrou um desempenho significativamente superior em comparação com um modelo de Rede Neural Convolutiva convencional. Uma segunda rede neural convolutiva, denominada "Rede de Controle", foi implementada para fins de comparação. Esta rede consiste em um módulo de extração (ResNet-50) e um módulo de decisão (Rede MLP), sem a utilização de Aprendizado por Múltiplas Instâncias ou o uso de um Módulo de Atenção. O Modelo de Controle gerou uma curva ROC com uma Área Sob a Curva (AUC) de 0.56, enquanto o modelo implementado neste trabalho obteve uma curva ROC com AUC de 0.82. Além disso, a ReceptorNet, implementada para realizar a mesma tarefa [9], alcançou uma curva ROC com AUC de 0.92.

Durante a elaboração deste projeto, diversas barreiras foram identificadas, limitando a obtenção de resultados ainda mais satisfatórios. Essas barreiras estão predominantemente associadas a limitações de hardware. Superar essas limitações torna viável a reprodução e aperfeiçoamento desse modelo em trabalhos futuros.

As imagens utilizadas para o treinamento do modelo ocupam 480.44 Gigabytes. Realizar todas as etapas de *data-augmentation* e pré-processamento implementados no artigo original exigem um *hardware* mais robusto. Além disso, o *Australian Breast Cancer Tissue Bank (ABCTB)* possui quase duas vezes mais imagens de amostras de tecido cancerígenos em Coloração H&E que podem ser utilizados para treinar a rede.

Outra limitação identificada está relacionada à dimensão dos slides utilizados no treinamento da rede. A utilização de uma resolução mais elevada pode resultar em melhorias nos resultados, no entanto, o aumento de um único nível de resolução implica em um aumento de aproximadamente 8 vezes na quantidade de *tiles* para cada slide. Isso demanda um espaço de armazenamento significativamente maior e a necessidade de uma placa gráfica mais robusta para lidar com o processamento correspondente.

Por fim, em todos os cenários implementados nesse trabalho, a ResNet-50 foi importada com as conexões entre os neurônios congeladas. Ajustar os pesos das camadas convolucionais durante o treinamento da rede aumentaria ainda mais a capacidade do Módulo de Extração de Características em se adaptar especificamente à tarefa de classificação de câncer de mama.

REFERÊNCIAS BIBLIOGRÁFICAS

- 1 SANTOS, M. d. O.; LIMA, F. C. d. S. d.; MARTINS, L. F. L.; OLIVEIRA, J. F. P.; ALMEIDA, L. M. d.; CANCELA, M. d. C. Estimativa de incidência de câncer no brasil, 2023-2025.
- 2 CANCER-DE-MAMA-RECEPTOR-DE-HORMONIO. <<http://www.oncoguia.org.br/conteudo/cancer-de-mama-receptor-de-hormonio/10879/264/#:~:text=Os%20c%C3%A2nceres%20receptores%20hormonais%20positivos,muitos%20anos%20ap%C3%B3s%20o%20tratamento>> Instituto Oncoguia - Câncer de Mama Receptor de Hormônio. Acessado: 2023-09-14.
- 3 HORMONIOTERAPIA no câncer de mama. <<https://www.einstein.br/especialidades/oncologia/exames-tratamentos/hormonioterapia-cancer-mama>>. Acessado: 2023-09-14.
- 4 MCCARTHY, J.; MINSKY, M. L.; ROCHESTER, N.; SHANNON, C. E. A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI Magazine*, p. 12, Dec. 2006.
- 5 SWANSON, J. A.; SCHMITZ, D.; CHUNG, K. C. How to practice evidence-based medicine. *Plast Reconstr Surg*, United States, v. 126, n. 1, p. 286–294, jul. 2010.
- 6 GROUP, E.-B. M. W. et al. Evidence-based medicine. a new approach to teaching the practice of medicine. *Jama*, v. 268, p. 2420–2425, 1992.
- 7 SUR, R. L.; DAHM, P. History of evidence-based medicine. *Indian J Urol*, India, v. 27, n. 4, p. 487–489, out. 2011.
- 8 MASIC, I.; MIKOVIC, M.; MUHAMEDAGIC, B. Evidence based medicine - new approaches and challenges. *Acta Inform Med*, Bosnia and Herzegovina, v. 16, n. 4, p. 219–225, 2008.
- 9 NAIK, N.; MADANI, A.; ESTEVA, A.; KESKAR, N. S.; Press, Michael F.; RUDERMAN, D.; AGUS, D. B.; SOCHER, R. Deep learning-enabled breast cancer hormonal receptor status determination from base-level H&E stains. *Nature Communications*, v. 11, n. 1, p. 5727, nov. 2020.
- 10 COUDRAY, N.; OCAMPO, P. S.; SAKELLAROPOULOS, T.; NARULA, N.; SNUDERL, M.; FENYÖ, D.; MOREIRA, A. L.; RAZAVIAN, N.; TSIRIGOS, A. Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning. *Nature Medicine*, v. 24, n. 10, p. 1559–1567, out. 2018.
- 11 YU, K.-H.; WANG, F.; BERRY, G. J.; RÉ, C.; ALTMAN, R. B.; SNYDER, M.; KOHANE, I. S. Classifying non-small cell lung cancer types and transcriptomic subtypes using convolutional neural networks. *J Am Med Inform Assoc*, England, v. 27, n. 5, p. 757–769, maio 2020.
- 12 WANG, D.; KHOSLA, A.; GARGEYA, R.; IRSHAD, H.; BECK, A. H. *Deep Learning for Identifying Metastatic Breast Cancer*. 2016.
- 13 CANCER - Screening and early detection. <<https://www.who.int/europe/news-room/fact-sheets/item/cancer-screening-and-early-detection-of-cancer>>. Accessed: 2023-10-17.
- 14 BREAST Cancer Early Detection and Diagnosis. <<https://www.cancer.org/cancer/types/breast-cancer/screening-tests-and-early-detection.html>>. Accessed: 2023-10-17.
- 15 MAGAKI, S.; HOJAT, S. A.; WEI, B.; SO, A.; YONG, W. H. An introduction to the performance of immunohistochemistry. *Methods Mol Biol*, United States, v. 1897, p. 289–298, 2019.

- 16 AMERICAN Society of Clinical Oncology/College of American Pathologists Guideline Recommendations for Immunohistochemical Testing of Estrogen and Progesterone Receptors in Breast Cancer. *Journal of Clinical Oncology*, v. 28, n. 16, p. 2784–2795, 2010. PMID: 20404251. Disponível em: <<https://doi.org/10.1200/JCO.2009.25.6529>>.
- 17 FELDMAN, A. T.; WOLFE, D. Tissue processing and hematoxylin and eosin staining. In: _____. *Histopathology: Methods and Protocols*. New York, NY: Springer New York, 2014. p. 31–43. ISBN 978-1-4939-1050-2. Disponível em: <https://doi.org/10.1007/978-1-4939-1050-2_3>.
- 18 COLORAÇÕES PARA MICROSCOPIA DE LUZ. <<https://mol.icb.usp.br/index.php/1-23-conceitos-basicos/>>. Accessed: 2023-10-19.
- 19 ALPAYDIN, E. *Introduction to Machine Learning*. 3. ed. Cambridge, MA: MIT Press, 2014. (Adaptive Computation and Machine Learning). ISBN 978-0-262-02818-9.
- 20 LECUN, Y.; BOSER, B.; DENKER, J. S.; HENDERSON, D.; HOWARD, R. E.; HUBBARD, W.; JACKEL, L. D. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, v. 1, n. 4, p. 541–551, 1989.
- 21 ZHANG, A.; LIPTON, Z. C.; LI, M.; SMOLA, A. J. *Dive into Deep Learning*. 2023.
- 22 NIELSEN, M. A. misc, *Neural Networks and Deep Learning*. Determination Press, 2018. Disponível em: <<http://neuralnetworksanddeeplearning.com/>>.
- 23 KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: PEREIRA, F.; BURGESS, C. J. C.; BOTTOU, L.; WEINBERGER, K. Q. (Ed.). *Advances in Neural Information Processing Systems* 25. Curran Associates, Inc., 2012. p. 1097–1105. Disponível em: <<http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>>.
- 24 SIMONYAN, K.; ZISSERMAN, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015.
- 25 MAXPOOL2D - PyTorch. <<https://pytorch.org/docs/stable/generated/torch.nn.MaxPool2d.html>>. Acessado: 2023-11-20.
- 26 TF.KERAS.LAYERS.MAXPOOLING2D. <https://www.tensorflow.org/api_docs/python/tf/keras/layers/MaxPooling2D>. Acessado: 2023-11-20.
- 27 MAXPOOLING2D layer-Keras. <https://keras.io/api/layers/pooling_layers/max_pooling2d/>. Acessado: 2023-11-20.
- 28 DENG, J.; DONG, W.; SOCHER, R.; LI, L.-J.; LI, K.; FEI-FEI, L. Imagenet: A large-scale hierarchical image database. In: IEEE. *2009 IEEE conference on computer vision and pattern recognition*. [S.l.], 2009. p. 248–255.
- 29 HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016. p. 770–778.
- 30 CLASSIFICATION: ROC Curve and AUC. <<https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc?hl=en>>. Accessed: 2023-12-06.
- 31 GENOMIC Data Commons Data Portal. <<https://portal.gdc.cancer.gov/>>. Accessed: 2023-12-02.
- 32 NATIONAL Cancer Institute. <<https://www.cancer.gov/>>. Accessed: 2023-12-02.
- 33 OTSU, N. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, v. 9, n. 1, p. 62–66, 1979.