UNIVERSIDADE ESTADUAL PAULISTA "JÚLIO DE MESQUITA FILHO"
INSTITUTO DE BIOCIÊNCIAS DE BOTUCATU

VICTOR GUSTAVO CAFERRO ARRUDA

Avaliação Comparativa de Modelos CNNs para Segmentação de Tumores Encefálicos: Aplicabilidade de Ferramentas XAI

Botucatu
2025

VICTOR GUSTAVO CAFERRO ARRUDA

Avaliação Comparativa de Modelos CNNs para Segmentação de Tumores Encefálicos: Aplicabilidade de Ferramentas XAI

Trabalho de conclusão de curso apresentada ao Instituto de Biociências de Botucatu - UNESP, como parte dos requisitos necessários para obtenção do título de Bacharel em Física Médica.

Orientador: Prof. Dr. José Luiz Rybarczyk Filho

Botucatu
2025

A779a

Arruda, Victor Gustavo Caferro
Avaliação Comparativa de Modelos CNNs para Segmentação de Tumores Encefálicos: Aplicabilidade de Ferramentas XAI / Victor Gustavo Caferro Arruda. -- Botucatu, 2025
42 f. : il., tabs.

Trabalho de conclusão de curso (Bacharelado - Física Médica) - Universidade Estadual Paulista (UNESP), Instituto de Biociências, Botucatu
Orientador: José Luiz Rybarczyk Filho

1. Segmentação semântica. 2. Diagnóstico de tumores encefálicos. 3. Ferramentas XAI. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Dados fornecidos pelo autor(a).

VICTOR GUSTAVO CAFERRO ARRUDA

AVALIAÇÃO COMPARATIVA DE MODELOS CNNs PARA SEGMENTAÇÃO DE TUMORES ENCEFÁLICOS: APLICABILIDADE DE FERRAMENTAS XAI

Trabalho de Conclusão de Curso, apresentado a Universidade Estadual Paulista, como parte das exigências para a obtenção do título de Bacharel, do curso de Graduação em Física Médica.

Local, 19 de novembro de 2025.
(data da defesa)

BANCA EXAMINADORA

_____
Prof. Dr. José Luiz Rybarczyk Filho
Departamento de Biofísica e Farmacologia
Instituto de Biociências de Botucatu - UNESP

_____
Profa. Dra. Daniela Renata Cantane
Departamento de Biodiversidade e Bioestatística
Instituto de Biociências de Botucatu - UNESP

_____
Prof. Dr. André Luis Debiaso Rossi
Departamento de Computação
Faculdade de Ciências – Câmpus Bauru - UNESP

À minha avó pelo suporte e apoio inestimável.
Aos meus familiares e amigos, que sempre estiveram presentes.
Ao meu orientador e professores, pelos ensinamentos transmitidos e auxílios que me capacitaram.

Agradecimentos

Inicialmente, agradeço à UNIVERSIDADE ESTADUAL PAULISTA "JÚLIO DE MESQUITA FILHO", pelo suporte através da Permanência Estudantil, que garantiu que eu pudesse continuar a frequentar a universidade, mesmo diante de todas as dificuldades.

Agradeço também aos recursos oferecidos pelo GridUnesp, de forma que "Esta pesquisa tornou-se possível graças aos recursos computacionais disponibilizados pelo Núcleo de Computação Científica (NCC/GridUNESP) da Universidade Estadual Paulista (UNESP)."

À minha avó, Cleonice Caferro, uma pessoa que sempre esteve ao meu lado e me apoiou, alguém que eu amo e que digo com uma grande certeza, que se não estivesse ao meu lado, eu não sei se seria capaz de estar aqui hoje.

Aos meus familiares, que também sempre me apoiaram e serviram como uma fonte de segurança, para que eu jamais desistisse.

Ao meu orientador, Prof. Dr. José Luiz Rybarczyk Filho, pelos ensinamentos, pelo tempo dedicado, por todo o apoio e suporte que me foi dado e principalmente por servir como uma pessoa de confiança.

E por fim, aos meus amigos, que passaram toda a graduação ao meu lado, oferecendo momentos alegres e às vezes um humor um pouquinho duvidoso, mas que certamente tornaram meus dias mais leves, principalmente em momentos que eu não me encontrava muito bem ou disposto. Vocês não sabem o quanto me fizeram bem, agradeço por tudo.

"A maneira mais segura de corromper um jovem é instruí-lo a ter em alta estima somente aqueles que pensam da mesma forma, e a rejeitar todos aqueles que pensam de forma diferente."

(Friedrich Nietzsche)

Resumo

A detecção manual de tumores encefálicos representa um grande desafio para os radiologistas, pois trata-se de uma tarefa complexa e demorada, na qual pode levar os profissionais à cometerem imprecisões no diagnóstico correto e, consequentemente, impactar negativamente o plano de tratamento do paciente. Atualmente, com o avanço de inteligências artificiais, o desenvolvimento e utilização de modelos de aprendizado profundo baseados em Redes Neurais Convolucionais (RNCs), observa-se uma eficiência significativa na segmentação, classificação e delimitação de bordas de tumores encefálicos a partir de imagens de ressonância magnética, otimizando a acurácia diagnóstica e a eficiência clínica, reduzindo de maneira significativa as imprecisões, erros e complicações que possivelmente poderiam ocorrer com a detecção manual desses tumores. Neste trabalho, abordamos a implementação de três diferentes modelos de segmentação semântica (SCU-Net_VGG16_ HDC, ResNet50_UNet e VGG19-UNet), sendo três modelos que ganharam um certo destaque nos últimos anos, devido às suas características individuais, de acordo com suas estruturas em relação aos dados de entrada e saída. O objetivo deste trabalho foi centralizado na realização de uma avaliação comparativa entre esses três modelos de segmentação, comparando a eficiência na segmentação e delimitação de bordas de tumores encefálicos, com a finalidade de otimizar a acurácia diagnóstica e a eficiência clínica. Essa comparação de desempenho foi realizada por meio de métricas padrões de avaliação de desempenho (*Dice Coefficient, Precision, Recall, F1-score*), função de perda ponderada *Weighted Tversky Loss*, ajustes e otimização de hiperparâmetros através do uso da biblioteca OPTUNA, e as explicações de decisões dos modelos foram realizadas por meio de uma Inteligência Artificial Explicável (XAI) com base no modelo *SHapley Additive exPlanations* (SHAP). Como esperado, os modelos de segmentação semântica investigados apresentaram uma grande eficiência, com leves variações nas segmentações de tumores encefálicos e em suas métricas de validação, devido aos ajustes e processos de otimização de hiperparâmetros dos modelos, que favoreceram para que os modelos treinados obtivessem uma alto nível de desempenho. A implementação de uma ferramenta explicativa, serviu como um critério adicional para segurança e confiabilidade nas tarefas de segmentação, visto que conseguiu indicar as regiões da imagem de maior contribuição para a decisão dos modelos em tarefas de segmentação e delimitação de bordas. O modelo ResNet50_UNet apresentou um desempenho levemente superior na segmentação e delimitação de bordas de tumores encefálicos, em relação aos modelos VGG19-UNet e SCU-Net_VGG16_ HDC, mas essas diferenças são mínimas e não impactam de maneira significativa em casos clínicos.

Palavras-chaves: Aprendizado Profundo, Redes Neurais Convolucionais (RNCs), Inteligência Artificial Explicável, avaliação comparativa, segmentação semântica.

Abstract

Manual detection of brain tumors represents a major challenge for radiologists, as it is a complex and time-consuming task that can lead professionals to make inaccuracies in the correct diagnosis and, consequently, negatively impact the patient's treatment plan. Currently, with the advancement of artificial intelligence, the development and use of deep learning models based on Convolutional Neural Networks (CNNs) shows significant efficiency in the segmentation, classification, and delimitation of brain tumor borders from magnetic resonance imaging, optimizing diagnostic accuracy and clinical efficiency, significantly reducing inaccuracies, errors, and complications that could possibly occur with the manual detection of these tumors. In this work, we address the implementation of three different semantic segmentation models (SCU-Net VGG16 HDC, ResNet50 UNet, and VGG19-UNet), three models that have gained prominence in recent years due to their individual characteristics, according to their structures in relation to input and output data. The objective of this work was to conduct a comparative evaluation between these three segmentation models, comparing the efficiency in segmentation and delimitation of brain tumor borders, in order to optimize diagnostic accuracy and clinical efficiency. This performance comparison was carried out using standard performance evaluation metrics (Dice Coefficient, Precision, Recall, F1-score), the Weighted Tversky Loss function, adjustments and optimization of hyperparameters through the use of the OPTUNA library, and the explanations of the models' decisions were performed using an Explainable Artificial Intelligence (XAI) based on the SHapley Additive exPlanations (SHAP) model. As expected, the investigated semantic segmentation models showed great efficiency, with slight variations in the segmentation of brain tumors and in their evaluation metrics, due to the adjustments and hyperparameter optimization processes of the models, which favored the trained models achieving a high level of performance. The implementation of an explanatory tool served as an additional criterion for safety and reliability in segmentation tasks, since it was possible to indicate the image regions that contributed most to the models' decision in segmentation and border delimitation tasks. The ResNet50 UNet model showed a slightly superior overall performance in the segmentation and border delimitation of brain tumors, compared to the VGG19-UNet and SCU-Net VGG16 HDC models, but these differences are minimal and do not significantly impact clinical cases.

Keywords: Deep Learning, Convolutional Neural Networks (CNNs), Explainable Artificial Intelligence (XAI), comparative evaluation, semantic segmentation..

Lista de figuras

Lista de tabelas

Lista de abreviaturas e siglas

RM	Ressonância Magnética
RNCs	Redes Neurais Convolucionais
SHAP	SHapley Additive exPlanations
AI	Artificial Intelligence
DL	Deep Learning
CNNs	Convolutional Neural Networks
XAI	EXplanable Artificial Intelligence
MRI	Magnetic Resonance Image
AATC	Associação Americana de Tumores Cerebrais
AAC	Associação Americana do Câncer
INCA	Instituto Nacional do Câncer
SNC	Sistema Nervoso Central
FNCs	full convolutional networks
FLAIR	Fluid Attenuated Inversion Recovery
HDC	Dilated Convolution
ReLu	Rectified Linear Unit
BN	Batch Normalization
PReLu	Parametric Rectified Linear Unit
DC	Dice Coeficient
TP	True Positive
FN	False Negative
FP	False Positive

Sumário

[1] De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

1 INTRODUÇÃO

1.1 Motivação

Os tumores encefálicos sempre foram temidos, visto que apresentam uma taxa elevada de incidência de 1,5% e uma taxa expressiva de mortalidade de 3% (ZHENG; ZHU; GUO, 2022). De acordo com a Associação Americana de Tumores Cerebrais (AATC), um tumor cerebral é identificado aproximadamente em cerca de 612.000 pessoas por ano na região do Ocidente (MUNOZ et al., 2014). Em 2021, a Associação Americana do Câncer (AAC) afirma que 78.980 pessoas foram diagnosticadas com tumores encefálicos, sendo 24.530 desses tumores classificados como malignos e 55.150 sendo considerados tumores não cancerosos, com uma incidência em 13.840 homens e 10.690 mulheres(MUNOZ et al., 2014). Alguns relatórios também indicam que a principal causa de mortalidade de tumores em crianças e adultos provém de tumores encefálicos(AYADI et al., 2021). Conforme um levantamento de dados realizado pelo Instituto Nacional do Câncer (INCA) no ano de 2022, o câncer no Sistema Nervoso Central (SNC), no qual incluem o cérebro e a medula espinhal, representa de 1,4% a 1,8% de todos os tumores malignos do mundo, sendo 88% desses casos no cérebro(Secretaria de Saúde do Distrito Federal, 2024). Ainda segundo o INCA, é estimado que aconteça uma incidência de cerca de 11.000 novos casos de tumores no SNC por ano no Brasil, com uma prevalência maior em homens (com mais de 6.000 casos)(Secretaria de Saúde do Distrito Federal, 2024).

Nesse contexto, o processo de segmentação de tumores encefálicos ou cerebrais a partir de exames médicos provenientes da Ressonância Magnética (RM) é fundamental para o diagnóstico, monitoramento e planejamento do plano terapêutico do paciente com condições neurológicas(PATRO; YADAV; YADAV, 2024). Essas tarefas de segmentação e delimitação de bordas desses tumores permitem que os médicos radiologistas consigam tomar decisões mais informadas e desenvolver planos de tratamento mais eficazes, assim como ter um melhor prognóstico e progressão da doença(PATRO; YADAV; YADAV, 2024).

Entretanto, mesmo com o avanço de tecnologias na área de imagens médicas, a segmentação manual de tumores encefálicos e sua delimitação de região tratam-se de tarefas complexas e demoradas, representando um grande desafio para os médicos radiologistas, visto que demanda uma certa expertise na realização destas tarefas, podendo resultar em imprecisões no diagnóstico e, consequentemente, resultar no desenvolvimento de um plano terapêutico não muito efetivo(SHARIF et al., 2024).

1.2 Redes neurais convolucionais (RNCs) e modelos de segmentação

Atualmente, modelos de aprendizado profundo baseados em redes neurais convolucionais (RNCs) têm demonstrado eficiência significativa na segmentação, classificação e delimitação de bordas de tumores encefálicos a partir de imagens de ressonância magnética, otimizando a acurácia diagnóstica e a eficiência clínica(ZHENG; ZHU; GUO, 2022).

No entanto, as estruturas de RNCs são propensas a processos que resultam em redundância computacional, devido ao processamento de muitas imagens de alta densidade, levando à necessidade de desenvolvimento de modelos de segmentação com estruturas U-Net, *Fully Convolutional Networks* (FCNs), e alguns outros algoritmos baseados em RNCs(YANG et al., 2020). Todavia, muitos desses algoritmos propostos apresentam problemas relacionados à precisão do modelo, extração de características ou classificações

de píxels, resultando em uma eficiência diagnóstica não muito apropriada(YANG et al., 2020).

Modelos de segmentação semântica pré-treinados baseados em estruturas U-Net, como o modelo SCU-Net, modelos ResNet baseados em redes residuais profundas e modelos híbridos como VGG-UNet, estão ganhando bastante notoriedade nestes últimos anos, devido às suas abordagens complementares, poder de extração de características e processamento com os dados de entrada (*input*) e dados de saída (*output*)(ZHENG; ZHU; GUO, 2022). Dessa forma, vem sendo observada uma melhora na precisão dos modelos, assim como uma redução do tempo de treinamento(ZHENG; ZHU; GUO, 2022).

Portanto, neste trabalho, foram implementados três modelos de segmentação que utilizam-se de redes neurais convolucionais para segmentação semântica de tumores encefálicos, devido às suas diferentes arquiteturas, complementariedade e tratamento de dados, sendo eles: SCU-Net_VGG16_HDC, VGG19-UNet e ResNet50_UNet. A arquitetura do modelo SCU-Net_VGG16_HDC é bastante precisa em tumores complexos e de bordas irregulares, devido a conexão entre dois módulos em série de codificação e decodificação, visto que o refinamento em duas etapas em relação às camadas de convoluções reduz erros de contorno e problemas de redundância, mas custa em velocidade de treinamento e processamento(ZHENG; ZHU; GUO, 2022). O modelo *VGG19-UNet* tem uma arquitetura profunda com 19 camadas convolucionais, apresentando alta extração de características e boa capacidade de segmentação em regiões amplas e homogêneas, porém é relativamente lenta e apresenta redundância de parâmetros(SIMONYAN; ZISSERMAN, 2014). Em relação à arquitetura do modelo ResNet50_UNet, ela apresenta uma alta capacidade de extração de características em regiões profundas e de segmentação de tumores em comparação com o modelo *VGG19-UNet*, assim como menor redundância, devido a presença de blocos residuais e 50 camadas convolucionais, sendo o modelo de segmentação mais equilibrado, combinando uma alta eficiência de segmentação, rapidez de treinamento e generalização de casos clínicos(HE et al., 2016a).

1.3 Ajustes de hiperparâmetros e aplicação de ferramentas XAI

No contexto clínico de diagnóstico, torna-se interessante maximizar o desempenho dos modelos na segmentação de máscaras tumorais encefálicas e delimitar de maneira mais precisa as bordas desses tumores. Com tal finalidade, é essencial a implementação de uma função de hiperparametrização para o treinamento dos modelos, garantindo uma maior confiabilidade dessas redes neurais convolucionais e algoritmos de aprendizado profundo supervisionados(ASIRI et al., 2024). Esses parâmetros exercem controle sobre algumas características da rede neural do modelo, abrangendo a extração de informações de acordo com as camadas convolucionais,a velocidade de convergência do modelo e sua complexidade(ASIRI et al., 2024).

O ajuste correto desses hiperparâmetros pode melhorar o desempenho da segmentação para que os modelos funcionem de maneira eficiente, sem perder informações e precisão na detecção de limites tumorais complexos em regiões de bordas, auxiliando a tomada de decisões precisas e otimizadas da equipe clínica no diagnóstico de tumores encefálicos(PAMUNGKAS et al., 2025).

Nesse sentido, ao utilizar sistemas de diagnósticos médicos baseados em Inteligência Artificial, assim como ao implementar ajustes de hiperparâmetros, torna-se fundamental avaliar e medir incertezas aleatórias e epistêmicas, que são os dois tipos básicos de incerteza em técnicas de aprendizado profundo, como medir a incerteza da predição do modelo e

as decisões tomadas durante a validação da saída de dados(ZEINELDIN et al., 2022). Consequentemente, o uso de ferramentas de Inteligência Artificial Explicável (XAI) recebeu certa relevância na aplicação e exploração da explicabilidade, com o intuito de descrever as tomadas de decisão de classificação e segmentação de tumores de forma compreensível. Essa explicabilidade garante uma maior confiança em termos de diagnóstico em técnicas de aprendizado profundo, além de servir como uma ferramenta auxiliar(ZEINELDIN et al., 2022).

De acordo com o Regulamento Geral de Proteção de Dados (RGPD) da União Europeia, torna-se imprescindível a explicação como um dos requisitos para algoritmos automatizados, para que sejam utilizados durante a prática clínica com pacientes(TEMME, 2017).

Apesar dos avanços na área de diagnóstico por imagens, através da utilização de algoritmos baseados em aprendizado profundo supervisionado, é visto uma necessidade de análise mediante a realização de avaliações comparativas sistemáticas que explorem eficiência, tempo de treinamento, limitações para tradução clínica e a utilização de métricas de avaliação além do *Dice coefficient* em relação aos modelos de segmentação, para que seja viável a integração desses modelos na prática clínica de diagnóstico de tumores encefálicos (TAVARES et al., 2024).

Portanto, este trabalho busca preencher essa lacuna científica no contexto de análises comparativas entre modelos, através da análise de desempenho por meio de métricas de avaliação e o estudo de suas decisões em tarefas de segmentação e delimitação de bordas realizado pela implementação de uma ferramenta explicável (XAI).

Por fim, este trabalho está organizado e estruturado da seguinte forma: No capítulo 2 encontra-se os objetivos de maneira detalhada, o capítulo 3 descreve detalhadamente o banco de dados utilizados e a metodologia seguida, o capítulo 4 apresenta os resultados obtidos e suas discussões, e por fim, o capítulo 5 conclui o trabalho, discutindo algumas implicações clínicas e perspectivas futuras.

2 OBJETIVOS

Objetivo geral

O objetivo geral deste trabalho está centralizado na proposta de realizar uma avaliação comparativa e estudo de decisões por meio de ferramentas explicativas (XAI) entre três diferentes modelos de aprendizado profundo, em tarefas de segmentação, classificação e delimitação de bordas de tumores encefálicos, devido as suas propriedades e arquiteturas únicas, assim como suas complementariedades no contexto de diagnóstico clínico e eficiência, sendo eles: SCU-Net_VGG16_HDC, VGG19-UNet e ResNet50_UNet.

Objetivos específicos

Os objetivos específicos são:

- Comparar a eficiência em relação à segmentação e delimitação de bordas em regiões de tumores encefálicos, por meio de métricas de validação dos modelos;
- Implementar ajustes de hiperparâmetros que maximizem o desempenho dos modelos em tarefas de segmentação;
- Utilizar técnicas de uma ferramenta de Inteligência Artificial Explicável (XAI), para interpretar as decisões dos modelos;
- Discutir a aplicabilidade clínica desses modelos no apoio ao diagnóstico de tumores encefálicos.

3 MATERIAIS E MÉTODOS

3.1 Fluxo de trabalho

Nesta seção, é apresentado um fluxograma representativo do fluxo de trabalho seguido, conforme apresentado na Figura 1, assim como uma descrição com uma sequência lógica de procedimentos para a conclusão do processo de treinamento e avaliação dos modelos.

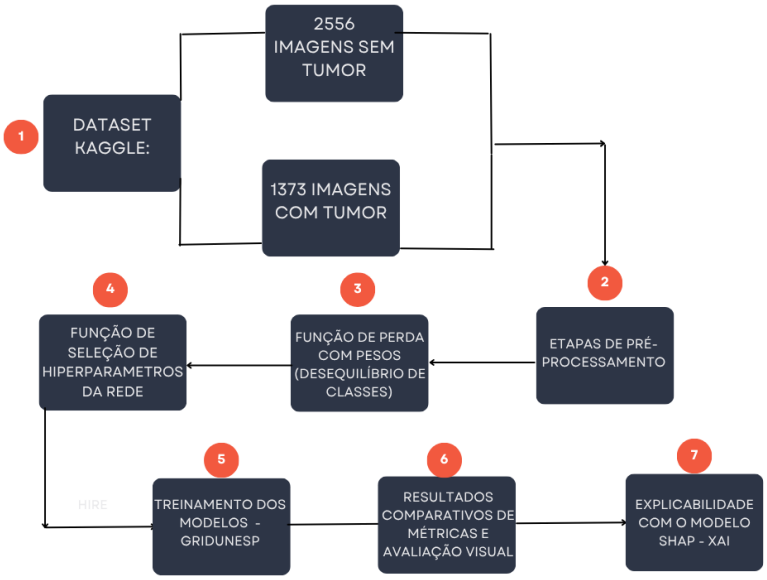


Figura 1 – Fluxograma representativo do fluxo de trabalho. Fonte: Próprio Autor.

Todas as etapas apresentadas no fluxograma estão descritas de maneira detalhada abaixo:

- Escolha do *dataset*, carregamento e realização de etapas de pré-processamento (pareamento, redimensionamento, normalização, divisão estratificada) nas imagens;
- Uso do sistema de *cluster* GridUnesp, visando a otimização de tempo na etapa de treinamento, devido ao acesso às *GPUs* e a alta capacidade de memória RAM, atingindo a ordem de *Terabytes*.
- Implementação das arquiteturas dos modelos trabalhados, e definição de métricas de avaliação de desempenho (*Dice Coefficient, F1-score, Recall, Precision, Weight Tversky Loss*;
- Ajustes e otimização de hiperparâmetros das redes neurais, com o objetivo de maximizar o desempenho dos modelos em tarefas de segmentação e delimitação de bordas;
- Definição e implementação de uma função *XAI* com o uso de ferramenta *SHAP* e *Overlays* para a explicabilidade das decisões dos modelos treinados;
- Como análise quantitativa, foram gerados gráficos comparativos entre as métricas de avaliação em relação a cada modelo (treinamento e validação) e uma tabela comparativa entre as melhores métricas para cada modelo. Como análise qualitativa e visual, foram geradas imagens comparativas entre as máscaras manuais e as

máscaras previstas por cada modelo após o treinamento, além de imagens visuais geradas pela ferramenta SHAPcom o uso de *Overlays* como suporte de avaliação.

3.2 Base de Dados

Neste trabalho, foi utilizado um *dataset* intitulado *"LGG MRI Segmentation"*, obtido no *Kaggle*, com imagens de Ressonância Magnética (RM) cerebral de 110 pacientes, contendo 2556 imagens sem tumor e 1373 imagens com tumor, com máscaras manuais de segmentação de anormalidades do tipo *Fluid Attenuated Inversion Recovery* (FLAIR), uma técnica de contraste de recuperação de inversão atenuada por fluido, que realça lesões como tumores encefálicos(BUDA, 2019).

3.2.1 Pré-processamento de dados

Foram realizadas algumas etapas de pré-processamento das imagens presentes neste *dataset*, conforme descrito a seguir:

- **Carregamento e pareamento das imagens**: Nesta etapa, o algoritmo percorre o *dataset* buscando arquivos em ".tif´", normalmente arquivos que contém máscaras de segmentação. Em seguida, ele separa as imagens de RM originais e suas máscaras correspondentes, com o objetivo de realizar um pareamento correto entre cada imagem e sua máscara a partir da numeração no nome do arquivo através da função *extract_number*. Por fim, constrói um *Dataframe* com três colunas principais: *image_path* (caminho da imagem de RM), *mask_path* (caminho da máscara manual) e *mask_status* (onde (1) indica a presença e (0) a ausência de tumor).
- **Redimensionamento e normalização:** Nesta etapa de pré-processamento, todas as imagens são redimensionadas para 256 x 256 pixels. Em seguida é realizada uma conversão de canais na imagem, ou seja, se a imagem está em **grayscale** (tons de cinza), ela é convertida para RGB (canais de vermelho, verde e azul), assim como para imagens RGBA que são convertidas para imagens RGB descartando o canal alfa. Por fim, é realizada uma normalização estatística nas imagens, por meio da subtração da média dos pixels da imagem e a divisão pelo desvio padrão (*standardization*), e assim, garantir uma estabilidade no treinamento. As máscaras também são redimensionadas pela função *cv2.resize* e são normalizadas para valores entre 0 e 1 ($mask/255$).
- **Divisão estratificada de dados:** A divisão estratificada de dados foi realizada da seguinte forma: 80% dos dados foram destinados para treinamento dos modelos, 10% para validação dos modelos e 10% para teste, para garantir que todos os subconjuntos tenham proporção balanceada de imagens com e sem tumor.

3.3 Arquitetura dos modelos de segmentação

3.3.1 Arquitetura e estrutura do modelo SCU-Net_VGG16_HDC

A arquitetura do modelo de rede de segmentação SCU-Net_VGG16_HDC é baseada na presença de dois módulos de codificação e decodificação, fazendo uma ponte de ligação entre suas camadas para fins de compartilhamento de informações e recursos, que são integrados em série como parte da estrutura básica do módulo de codificação a rede neural VGG16 (como um encoder pré-treinado) e o modelo *Hybrid Dilated Convolution* (HDC)(ZHENG; ZHU; GUO, 2022). De acordo com alguns estudos, essas interações entre camadas auxiliam na extração de características e nas tarefas de segmentação semântica(ZHENG; ZHU; GUO, 2022). Em geral, a grande maioria dos modelos de redes neurais, utilizam um agrupamento máximo para reduzir o volume da rede e destacar de maneira mais evidenciada as principais características extraídas durante as etapas de convolução para extração de características(ZHENG; ZHU; GUO, 2022). A principal problemática ao utilizar-se desse método, está relacionada de forma direta ao fato de que os detalhes de segmentação podem ser ignorados e as informações de localização espacial dessas principais características podem ser perdidas(ZHENG; ZHU; GUO, 2022). Porém, os cortes de tumores encefálicos requerem uma precisão em escala milimétrica, ou seja, infere-se a necessidade de informações mais detalhadas e a minimização da perda de características durante o treinamento.

De acordo com a literatura, sabe-se que *kernels* convolucionais maiores conseguem extrair mais informações posicionais de maneira mais precisa do que *kernels* convolucionais menores, devido à presença de um campo receptivo maior, aumentando a resolutividade dessas informações posicionais(ISLAM; JIA; BRUCE, 2020).

Nesse sentido, o modelo SCU-Net_VGG16_HDC, com a implementação de uma rede VGG16, atuando como um *encoder* pré-treinado e com a integração de um módulo HDC, permite a otimização e maior eficiência durante o treinamento, o aumento do campo receptivo da rede de codificação, aprimorando a capacidade de captura de detalhes da imagem de entrada, de informações de borda e da posição relativa do tumor, além de reduzir o problema da grade de convolução (WANG et al., 2018).

O operador do módulo HDC é apresentado e definido na equação 1 como:

$$\begin{aligned}
(F *_{l_1} k)(\mathbf{p}) &= \sum_{s+l_1 t=\mathbf{p}} F(s)k(t) \\
(F *_{l_2} k)(\mathbf{p}) &= \sum_{s+l_2 t=\mathbf{p}} F(s)k(t) \\
&\vdots \\
(F *_{l_n} k)(\mathbf{p}) &= \sum_{s+l_n t=\mathbf{p}} F(s)k(t)
\end{aligned} \tag{1}$$

Sendo que, na equação 1, F trata-se do mapa de características (*Feature map*), k representa o *kernel* ou filtro discreto, s representa a posição do pixel, t é a posição do peso dentro do *kernel*, ln e $l2$ são operadores de convolução dilatada híbrida com diferentes fatores de dilatação. Na estrutura de decodificação do modelo SCU-Net_VGG16_HDC, cada convolução dilatada híbrida e uma função ReLU, que é uma função linear de ativação

retificadora, formam um módulo HDC, e três módulos HDC com diferentes taxas de expansão formam um grupo de módulos HDC(ZHENG; ZHU; GUO, 2022). Essa estrutura da rede de codificação é dividida em cinco camadas, apresentada na Figura 14, anexada no Apêndice A.

Cada camada presente na rede de codificação é composta por um agrupamento máximo e um grupo de blocos de convolução, sendo que cada grupo desses blocos apresentam três convoluções dilatadas 3×3 (com taxas de dilatação 1, 2 e 3), além de três camadas de *Batch Normalization* (BN) e três funções ReLU (ZHENG; ZHU; GUO, 2022).

O módulo de estrutura da rede de decodificação do modelo SCU-Net_VGG16_HDC é composto por múltiplas convoluções e convoluções transpostas, em que as informações relacionadas à camada seguinte são expandidas por uma convolução transposta de tamanho 3×3 pixels e, em seguida, são concatenadas e convoluídas com seu respectivo mapa de características(ZHENG; ZHU; GUO, 2022). Caso essas informações de camada também estiverem na segunda rede de decodificação, elas também precisam ser convoluídas com a imagem da mesma camada após a etapa de primeira decodificação, depois ampliadas e concatenadas com a camada superior, e de maneira repetida, passando por processos de ampliação, concatenação e convolução, recuperando assim continuamente os pixels até que se tornem iguais à imagem de entrada e, por fim, passando por uma nova convolução, de modo que o número de canais da imagem seja o número da classificação desejada(ZHENG; ZHU; GUO, 2022). Tal método de decodificação, ao combinar todas as informações obtidas em camadas anteriores e com estruturação em série de redes, favorece a obtenção de informações semânticas, além de conseguir extrair efetivamente as características de mesma camada diversas vezes, resultando em uma melhor previsão e capacidade, tornando o modelo mais preciso em relação às tarefas de segmentação e delimitação de bordas(ZHENG; ZHU; GUO, 2022).

A Figura 15, presente no Apêndice A, mostra a arquitetura geral do modelo SCU-Net_VGG16_HDC com suas estruturas de duas redes de codificação-decodificação, assim como várias operações de concatenação entre essas duas séries.

Diferentemente da rede U-Net, o modelo SCU-Net_VGG16_HDC utiliza diretamente a saída da estrutura dos módulos de codificação-decodificação anterior como uma entrada da próxima estrutura de codificação-decodificação, diminuindo e evitando a redundância estrutural e sendo mais efetiva em relação à extração de informações-chaves e mineração, além de maximizar o uso das informações obtidas da imagem após a decodificação anterior(ZHENG; ZHU; GUO, 2022). Essa conexão entre as mesmas camadas de duas estruturas de codificação-decodificação por meio do processo de concatenação, aprofunda a interconexão entre essas duas redes, aumentando a eficiência do modelo, otimizando o compartilhamento de informações semânticas e reduzindo a redundância de recursos, assim como reduzir o tempo de processamento dessas informações(ZHENG; ZHU; GUO, 2022).

3.3.2 Arquitetura do modelo *VGG19-UNet*

Na arquitetura do modelo VGG19-UNet, a rede VGG19 pré-treinada no conjunto de dados *ImageNet*, atua como parte codificadora da estrutura, extraindo características hierárquicas das imagens de entrada, enquanto a arquitetura U-Net atua como parte decodificadora, realizando um refinamento nas máscaras de segmentação e combinando extrações de características de baixo e alto nível(PATRO; YADAV; YADAV, 2024). Devido a complementariedade de suas estruturas unificadas, o modelo híbrido VGG19-UNet realiza a

segmentação de maneira precisa, além de minimizar a complexidade computacional(PATRO; YADAV; YADAV, 2024).

A parte codificadora e decodificadora do modelo modelo VGG19-Unet é descrita da seguinte forma:

- **Codificador *VGG19*:** Trata-se de uma rede neural convolucional profunda que contém em sua estrutura 16 camadas convolucionais e três camadas totalmente conectadas, conforme a figura 16, anexada no Apêndice A. As camadas convolucionais da parte codificadora VGG19 são responsáveis em capturar características complexas das imagens de entrada, enquanto as camadas toltalmente conectadas são removidas, visto que elas não são adequadas para tarefas de segmentação em nível de pixels (POURMAHBOUBI; SAEED; TABRIZCHI, 2025). Ambas as partes codificadora e decodificadora apresentam blocos convolucionais com suas respectivas camadas, seguidas por funções de normalização e ativação ReLu. A função de ativação em lote faz a padronização de entrada para cada camada convolucional, aumentando sua convergência e produzindo uma generalização mais aprimorada, devido a redução de deslocamentos internos de covariáveis(POURMAHBOUBI; SAEED; TABRIZCHI, 2025). A função de ativação *Rectified Linear Unit* (ReLu) introduz o conceito de não linearidade, no qual permite que o modelo faça a captura de padrões complexos na base de dados. Já a camada final da estrutura realiza uma operação de convolução com uma função de ativação *Sigmoide*, que calcula e indica a probabilidade de cada pixel da imagem pertencer a uma classe tumoral ou não, resultando em uma máscara de segmentação binária(POURMAHBOUBI; SAEED; TABRIZCHI, 2025).
- **Decodificador U-Net:** Em relação ao componente decodificador, a estrutura U-Net é eficiente na segmentação semântica no âmbito de imagens biomédicas. Sua arquitetura, presente na Figura 17, anexada no Apêndice A, possui um caminho de contração que reduz as dimensões espaciais dos mapas de características, sendo que ao mesmo tempo captura informações contextuais, e um caminho expansivo que utiliza-se de processos de *Upsampling* que restaura a resolução espacial da imagem, realizando um refinamento na máscara de segmentação(PATRO; YADAV; YADAV, 2024). O caminho de contração possui camadas convolucionais e de agrupamentos, que extraem características hierárquicas das imagens de entrada e as transmitem ao caminho expansivo, que por meio de conexões de salto é permitida a integração de características de baixo e alto nível para obter segmentações mais precisas(PATRO; YADAV; YADAV, 2024).

  Os mapas de características de saída da parte codificadora VGG19 servem como entrada para a parte decodificadora U-Net, que posteriormente passam por operações de *Upsampling* e concatenação para produção das máscaras finais. Dessa forma, essa integração favorece o equilíbrio entre a extração de características discriminativas nas imagens e a preservação das informações espaciais(PATRO; YADAV; YADAV, 2024).

3.3.3 Arquitetura geral da rede ResNet e estrutura do modelo ResNet50_UNet

A estrutura do modelo ResNet é baseada na adição de blocos residuais em vez dos blocos convolucionais presentes na estrutura U-Net original, sendo que cada um desses blocos residuais possui duas unidades convolucionais e em cada unidade incluem uma

camada de *Batch Normalization* (BN), uma função de ativação chamada de *Parametric Rectified Linear Unit* (PReLu) e uma camada convolucional(HE et al., 2016b).

Em RNCs profundas, a problemática do gradiente surge durante o processo de treinamento da rede neural, quando a norma do gradiente das camadas anteriores reduzem para zero, e como solução desta problemática, o modelo de aprendizagem residual da ResNet é implementado(EBRAHIMI; ABADI, 2018). Na rede residual ResNet, o rendimento de cada camada residual passa por uma convolução juntamente com sua entrada, para que seja a entrada da próxima camada. Um mapeamento residual $H(x)$ é utilizado para construir um bloco de aprendizagem residual da rede. Este bloco calcula de maneira aproximada $H(x) = F(x)+x$, no qual essa formulação é reconhecida por sistemas neurais com conexões de atalho que combinam a entrada e saída da camada empilhada por meio de uma operação de mapeamento sem adicionar nenhum parâmetro(IOFFE; SZEGEDY, 2015). Dessa forma, as normas dos gradientes não reduzem para zero e os gradientes podem fluir de forma facilitada, resultando em um tempo de treinamento mais rápido e a presença de um número maior de camadas na rede(HE et al., 2015).

O modelo ResNet consiste em dois principais blocos, sendo eles: Blocos de identidade e Blocos convolucionais.

- **Bloco de Identidade:** O bloco de identidade utilizado no modelo ResNet é definido na equação 2:

$$y = F(x, Wi) + x, \tag{2}$$

  onde **x** representa vetores de entrada e **y** são vetores de saída das camadas consideradas.
  A função $F(x, Wi)$ representa o mapeamento residual treinado, de modo que o bloco de identidade apresenta a mesma dimensão de x e F(IOFFE; SZEGEDY, 2015). Na Figura 18, presente no Apêndice A, é mostrada a estrutura do bloco de identidade e seus três componentes.
- **Bloco Convolucional:** No bloco convolucional, diferentemente do bloco de identidade, as dimensões de entrada e saída não são correspondentes, necessitando que as conexões de atalho executem uma projeção linear W_s para redimensionar as dimensões entre x e F(SHEHAB et al., 2021). A função para o bloco convolucional é dada conforme a equação 3:

$$y = F((x, Wi) + W_s x), \tag{3}$$

  Em que F é a saída da camada empilhada, x é o vetor de entrada e y o vetor de saída combinado do bloco convolucional.
  A estrutura do bloco convolucional segue o mesmo procedimento presente no bloco de identidade, porém com a adição de uma camada convolucional 2D juntamente ao atalho(SHEHAB et al., 2021). Neste atalho modificado, os vetores de entrada x são redimensionados para corresponder à saída do caminho principal, permitindo que o modelo adquira o conhecimento de uma certa função de identidade e garantindo que a camada superior tenha desempenho nas tarefas de mesma classe que as camadas inferiores(IOFFE; SZEGEDY, 2015).

A Figura 19, anexada no Apêndice A, apresenta a diferença entre a estrutura do bloco convolucional em relação ao bloco de identidade pela adição de uma camada convolucional 2D.

De acordo com (IOFFE; SZEGEDY, 2015), os autores descreveram diferentes configurações de modelos ResNets com 18, 34, 50, 101 e 152 camadas. Dessa forma, os blocos de identidade e de convolução são combinados para estruturar um modelo ResNet, neste caso sendo o modelo ResNet50_UNet (SHEHAB et al., 2021). A arquitetura do modelo ResNet50_UNet é apresentada e detalhada na Figura 20, presente no Apêndice A.

3.3.4 Treinamento dos modelos

Os treinamentos dos modelos foram realizados em um sistema de cluster, chamado GridUnesp, baseado no fluxo simultâneo de computadores com um volume de trabalho grande e de maneira repetida, com a integração de vários clusters que operam de forma coordenada e interligada, realizando a paralelização de serviços ou tarefas(Núcleo de Computação Científica (NCC), UNESP, 2022). A necessidade de realizar este trabalho neste sistema de cluster foi devido a integração e uso de GPUs e recursos computacionais oferecidos, visando a otimização do tempo de treinamento dos modelos e paralelização de tarefas no algoritmo utilizado.

3.3.5 Função de hiperparametrização: Otimização e escolha de hiperparâmetros dos modelos

A otimização (processo de ajuste de parâmetros internos do modelo (pesos e vieses) com o intuito de minimizar uma função de perda predefinida) e ajuste de hiperparâmetros (subárea da otimização que lida com as configurações de hiperparâmetros de alto nível que controlam como um modelo aprende) em Redes Neurais Convolucionais (RNCs) desempenham um papel essencial em relação ao desempenho do modelo em tarefas de segmentação, estabilidade de treinamento e sua capacidade de generalização de informação para novos dados(RAIAAN et al., 2024). Os principais hiperparâmetros em RNCs que afetam significativamente o desempenho dos modelos de aprendizado profundo e sua eficácia de treinamento, incluem a taxa de aprendizado do modelo (α), tamanho do lote, número de épocas de treinamento, otimizador e a função de ativão definida(MEZZAH; TARI, 2023).

A taxa de aprendizado α trata-se de um parâmetro que define a intensidade que o modelo atualiza os seus pesos durante cada iteração no seu treinamento, porém uma taxa de aprendizado muito alta pode resultar em uma convergência instável, fazendo com que o modelo ultrapasse o ponto ótimo durante o processo de otimização(PAMUNGKAS et al., 2025). Entretanto, uma taxa de aprendizado muito baixa pode resultar em um treinamento muito lento ou estagnar o modelo em um mínimo local(PAVIGLIANITI et al., 2022). Desse modo, a taxa de aprendizado normalmente é definida entre o intervalo de 0.0001 a 0.01, no qual varia dependendo da arquitetura do modelo(PAMUNGKAS et al., 2025).

O tamanho do lote trata-se de outro hiperparâmetro essencial, visto que ele determina o número de amostras de dados processadas antes da atualização dos pesos do modelo em uma única iteração e sua escolha afeta de maneira significativa a estabilidade e eficiência do treinamento(PAMUNGKAS et al., 2025). Em relação ao seu tamanho

(número de instâncias de dados), a escolha de um tamanho de lote pequeno, geralmente de tamanho 16 ou 32, se traduz em atualizações de pesos mais frequentes, ajudando o modelo a encontrar soluções ótimas mais rapidamente, porém necessita de mais iterações(HWANG et al., 2024). Um tamanho de lote maior, geralmente de tamanho 128 ou 256, resulta na utilização de mais dados antes da atualização dos pesos, melhorando a estabilidade do treinamento, mas exige mais memória em GPU(HWANG et al., 2024). Portanto, o tamanho do lote geralmente é determinado através de testes, pois um tamanho de lote muito grande pode resultar na redução da generalização do modelo, enquanto um tamanho de lote muito pequeno pode resultar em um treinamento mais ruidoso e instável(KANDEL; CASTELLI, 2020).

Outro parâmetro relevante relacionado ao desempenho dos modelos de RNCs trata-se do número de épocas definidas, no qual refere-se ao número de vezes que todo o conjunto de dados definido é utilizado para treinar o modelo(PAMUNGKAS et al., 2025). Caso seja definido um número de épocas muito baixo (geralmente entre 1 e 50 épocas), pode resultar em um subajuste, levando a aprendizagem não eficiente do modelo, mas ao se definir um número de épocas muito alto (geralmente acima de 50 épocas), o modelo pode ser induzido à memorização dos dados de treinamento e resultar em um superajuste, ocasionando um baixo desempenho em relação a novos dados(XU; COEN-PIRANI; JIANG, 2023). Nesse contexto, com o intuito de evitar problemas relacionados ao superajuste, técnicas de parada antecipada ou *Early Stopping* são utilizadas para interromper o processo de treinamento quando não é observada nenhuma melhora no desempenho do modelo ao longo de várias épocas(PAMUNGKAS et al., 2025). A escolha do número de épocas geralmente depende do conjunto de dados trabalhado e da complexidade do modelo escolhido, porém normalmente é definido dentro do intervalo de 10 a 100 épocas(AHMED et al., 2023).

Um outro aspecto relevante no processo de treinamento de RNCs é a escolha de um otimizador, que com base em cálculos de gradiente, ele determina de que forma o modelo atualiza os seus pesos durante seu treinamento(SHAO et al., 2024). A escolha desse otimizador normalmente também depende do conjunto de dados e da arquitetura de rede utilizada, sendo que o otimizador Adam é amplamente utilizado no contexto de imagens biomédicas para tarefas de segmentação e classificação, devido ao seu ótimo desempenho em redes neurais profundas, pois ajusta de forma dinâmica a taxa de aprendizado para cada parâmetro da rede, acelerando sua convergência(PAMUNGKAS et al., 2025).

Por fim, a definição de uma função de ativação em RNCs desempenha um papel essencial na introdução de não linearidade na rede, permitindo que o modelo aprenda padrões complexos de informação(ZHAO et al., 2024). Essa definição de uma função de ativação depende da tarefa realizada, sendo que a função ReLU é usualmente utilizada em camadas ocultas, devido a sua simplicidade e eficiência em resolver o problema de gradiente evanescente(NOEL et al., 2025).

Portanto, neste trabalho foi implementado um algoritmo que realiza a seleção de hiperparâmetros durante a etapa de treinamento, descrito no algoritmo 1, e um algoritmo que realiza um processo de otimização de hiperparâmetros, apresentado no algoritmo 2, através de métodos de otimização Bayesiana, assim como métodos Heurísticos, com o intuito de maximizar o desempenho dos modelos de acordo com os recursos computacionais requisitados. Os algoritmos 1 e 2 estão definidos e anexados no Apêndice B.

3.3.6 Métricas de avaliação

Neste trabalho, foram utilizadas as seguintes métricas de avaliação e perda para analisar o desempenho de modelos de redes neurais convolucionais: *Dice Coefficient* (DC), *Precision*, *F1-score*, *Recall* e *Weighted Tversky Loss*. Essas métricas foram escolhidas para avaliação dos modelos, pois são métricas bem consolidadas e comumente utilizadas para análise de performance de redes neurais convolucionais em tarefas de segmentação e delimitação de bordas, (CHOWDHURY; SRIVASTAVA, 2024).

As métricas de avaliação e perda estão definidas e detalhadas abaixo:

- *Dice Coefficient:* O *Dice Coefficient* mede e indica o grau de similaridade entre a máscara prevista pelo modelo e a máscara manual segmentada, variando entre o intervalo de 0 a 1, com o valor 1 indicando uma sobreposição perfeita entre as máscaras(CHOWDHURY; SRIVASTAVA, 2024), sendo definido de acordo com a equação 4:

$$DC = \frac{2TP}{FP + 2TP + FN} \tag{4}$$

  em que, TP (True positive) é o número de pixels tumorais reconhecidos com precisão, enquanto FP (False Positive) é o número de pixels não tumorais distinguidos de maneira efetiva e FN (False Negative) representa o número de pixels não tumorais identificados incorretamente.
- *Precision:* A métrica *Precision* indica a proporção de pixels TP entre todos os pixels que foram previsto como positivos(CHOWDHURY; SRIVASTAVA, 2024), definida abaixo na equação 5:

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

- *Recall:* A métrica *Recall* mensura a proporção de pixels positivos verdadeiros entre todos os pixels positivos reais classificados(CHOWDHURY; SRIVASTAVA, 2024), sendo calculada a partir da equação 6

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

- *F1-score:* A métrica *Precision* trata-se da média harmônica entre as métricas *Recall* e *Precision*, fornecendo uma métrica que equilibra ambas(CHOWDHURY; SRIVASTAVA, 2024), definida na equação 7:

$$F1\text{-}score = \frac{2X\,Precision X\,Recall}{Precision\ +\ Recall} \tag{7}$$

- *Weighted Tversky Loss:* A métrica de perda *Weighted Tversky Loss* é uma variação da métrica *Tversky Loss* utilizada em (ABRAHAM; KHAN, 2019), que produz uma função de perda que combina o índice *Tversky* com pesos definidos e diferentes para FNs e FPs, com um termo focal exponencial que prioriza o aprendizado em exemplos difíceis, ou seja, uma métrica ideal para segmentação desbalanceada onde FNs são caros. A sua equação completa é definida na equação 8:

$$TI_w = \frac{TP + \text{smooth}}{TP + \alpha \cdot w_{FN} \cdot FN + (1 - \alpha) \cdot w_{FP} \cdot FP + \text{smooth}} \tag{8}$$

onde:

- TP — número de pixels corretamente classificados como positivos.
- FN — número de pixels positivos incorretamente classificados como negativos.
- FP — número de pixels negativos incorretamente classificados como positivos.
- α — parâmetro que controla a penalização dos falsos negativos.
- w_{FN} — peso adicional aplicado aos falsos negativos.
- w_{FP} — peso adicional aplicado aos falsos positivos.
- smooth — termo de suavização numérica pequeno ($\approx 10^{-5}$) para evitar divisão por zero.

A equação 9 abaixo, define uma função de perda ponderada com pesos, que apresenta um termo de foco exponencial com um parâmetro γ que controla o nível de ênfase em exemplos difíceis:

$$L_{WFTL} = (1 - TI_w)^{\gamma} \tag{9}$$

- γ — parâmetro de foco (*focal exponent*) que controla o quanto os exemplos difíceis são enfatizados.
  - $\gamma > 1$ aumenta a ênfase em exemplos difíceis.
  - $\gamma = 1$ reduz a função para a Tversky Loss tradicional.

3.3.7 Ferramenta explicativa XAI - SHAP

Atualmente, algumas ferramentas de explicabilidade como métodos XAI que lidam com diversos dados e tipos de modelos têm sido abordadas, com o objetivo de explicar as saídas dos modelos de forma local e global (SALIH et al., 2025). Entre essas ferramentas, destacam-se as ferramentas *Shapley Additive Explanations* (SHAP) e *Local Interpretable Model Agnostic Explanation* (LIME), com abordagens em diferentes domínios de explicabilidade(SALIH et al., 2025).

O modelo SHAP trata-se de um método agnóstico de modelos *post-hoc*, no qual baseia-se na teoria dos jogos, calculando a contribuição de cada recurso do modelo, considerando todas as combinações entre as características extraídas e seu subconjunto de características que são usadas no modelo(SALIH et al., 2025). Portanto, o SHAP é um método de explicabilidade dependente do modelo, ou seja, seu resultado de valores Shapley depende do modelo utilizado para tarefas de classificação/regressão, possivelmente resultando em diferentes pontuações de explicabilidade entre os diferentes modelos para o mesmo conjunto de imagens(SALIH et al., 2025).

No contexto de aprendizado de máquina, cada pixel da imagem x_i contribui para o resultado de predição do modelo f(x), onde os valores Shapley, definidos como a contribuição marginal média da característica i em todas as possíveis combinações de subconjuntos de variáveis(LUNDBERG; LEE, 2017), são representados e definidos pela equação 10:

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(M - |S| - 1)!}{M!} \left[f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S) \right] \tag{10}$$

Em que:

- ϕ_i é o valor de Shapley da característica i;

- F é o conjunto total de características do modelo;
- S é um subconjunto de F que não contém a variável i;
- M é o número total de características;
- $f_S(x_S)$ representa a saída do modelo considerando apenas as características em S;
- $f_{S\cup\{i\}}(x_{S\cup\{i\}})$ inclui a característica i.

De modo que, o modelo SHAP satisfaz as seguintes propriedades definidas abaixo(LUNDBERG; LEE, 2017):

1. **Acurácia local:** a soma das contribuições que realiza a explicação da predição do modelo é calculada de acordo com a equação 11:

$$f(x) = f(x') + \sum_{i=1}^{M} \phi_i \tag{11}$$

2. **Consistência:** se a contribuição de uma variável aumenta, seu valor de Shapley não sofre uma diminuição.
3. **Ausência:** características ausentes têm valor de Shapley de $\phi_i = 0$.

Em modelos de classificação e segmentação complexos, como redes neurais profundas, os cálculos de maneira exata para as combinações ϕ_i são inviáveis, pois requer um número muito alto de avaliações referentes aos modelos(CHEN; LUNDBERG; LEE, 2020). Nesse sentido, o modelo *DeepSHAP* aproxima esses valores combinando a estrutura do modelo SHAP com o mecanismo de retropropagação de relevância do *DeepLIFT*, propagando as contribuições de forma eficiente camada a camada, devido a linearidade presente em redes neurais convolucionais(CHEN; LUNDBERG; LEE, 2020). Dessa forma, a relevância total referente a uma entrada i é obtida a partir da soma das contribuições das ativações j, conforme a equação 12:

$$\phi_i = \sum_j R_{ij} \tag{12}$$

Em que a relevância R_{ij} é calculada pela equação 13:

$$R_{ij} = \frac{(x_i - x'_i)w_{ij}}{\sum_k (x_k - x'_k)w_{kj}} R_j \tag{13}$$

sendo que:

- x_i: valor do recurso de entrada;
- x'_i: valor do recurso de referência;
- w_{ij}: peso para a conexão dos neurônios i e j;
- R_j: relevância acumulada no neurônio da camada superior.

Portanto, no contexto deste trabalho, em segmentação de imagens biomédicas, o *DeepSHAP* gera mapas de relevância espacial, permitindo visualizar exatamente quais regiões de pixels da imagem contribuíram mais fortemente para a identificação das estruturas segmentadas, a partir da sobreposição das máscaras segmentadas com *SHAPS overlays*.

4 RESULTADOS E DISCUSSÃO

4.1 Resultados

Os resultados para avaliação e explicação dos modelos de segmentação treinados foram divididos em dois tipos de análises, sendo elas:

- **Análise quantitativa e gráfica:** Resultados quantitativos obtidos por meio de gráficos comparativos entre cada métrica de avaliação em função das épocas definidas, tanto para a etapa de validação, quanto para a etapa de treino, apresentados nas figuras 2 à figura 11, assim como tabelas que apresenta os melhores valores de cada métrica de avaliação referente a cada modelo de segmentação, apresentados na tabela 1 e tabela 2.
- **Análise qualitativa e visual:** Resultados qualitativos e visuais obtidos através de uma imagem comparativa entre as máscaras reais segmentadas e as máscaras previstas por cada modelo após o treinamento e validação, apresentada na figura 12, assim como imagens explicativas geradas pelo modelo *SHAP*, uma ferramenta *XAI* que produz imagens com gradientes de calor e valores *Shapley* indicando quais regiões de pixels contribuiram mais para a decisão do modelo nas tarefas de segmentação e delimitação de bordas de tumores encefálicos, apresentadas na figura 13.

4.1.1 Análise quantitativa e gráfica do desempenho dos modelos

Os resultados quantitativos estão apresentados abaixo, nas figuras 2 à figura 11:

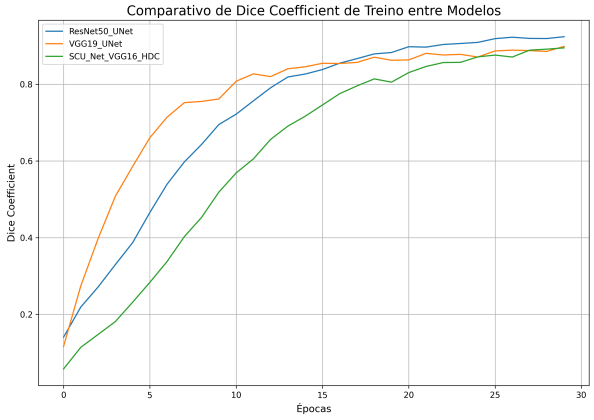


Figura 2 – Gráfico comparativo da métrica *Dice* durante a etapa de treino para cada modelo. Fonte: Próprio Autor.

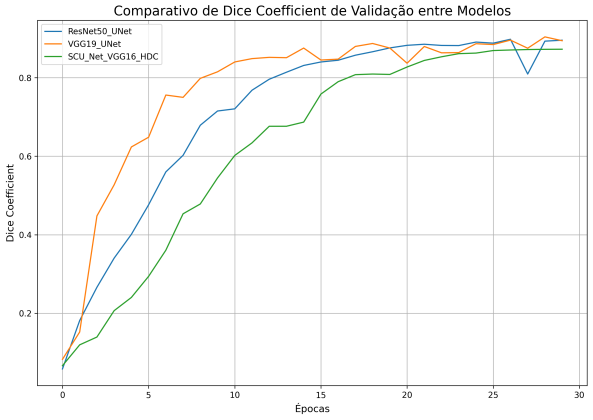


Figura 3 – Gráfico comparativo da métrica *Dice* durante a etapa de validação para cada modelo. Fonte: Próprio Autor.

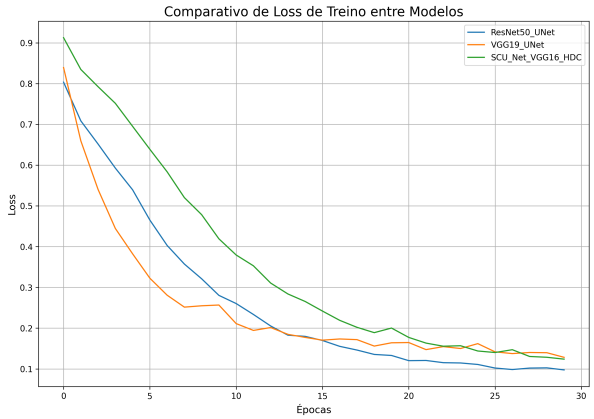


Figura 4 – Gráfico comparativo relacionado a métrica *Weighted Tversky Loss* durante a etapa de treino para cada modelo. Fonte: Próprio Autor.

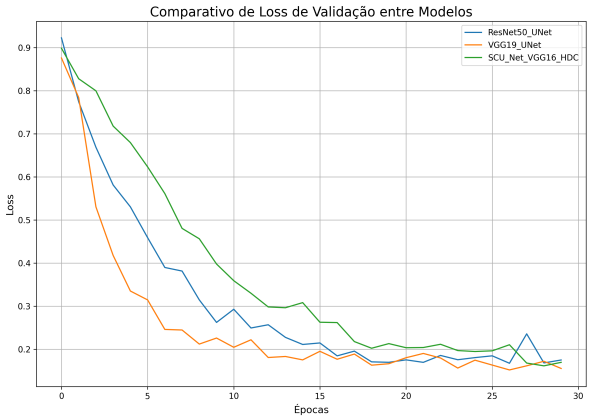


Figura 5 – Gráfico comparativo relacionado a métrica *Weighted Tversky Loss* durante a etapa de validação para cada modelo. Fonte: Próprio Autor.

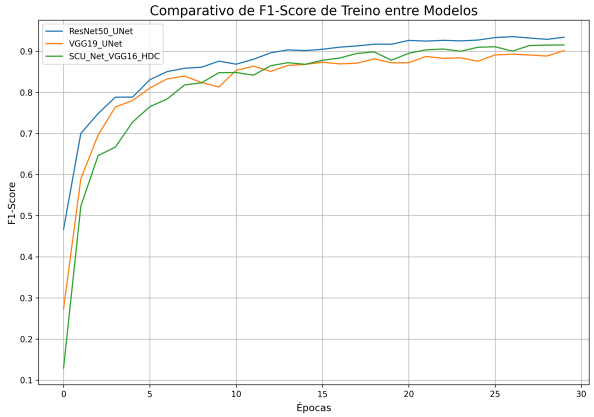


Figura 6 – Gráfico comparativo relacionado a métrica *F1-score* durante a etapa de treino para cada modelo. Fonte: Próprio Autor.

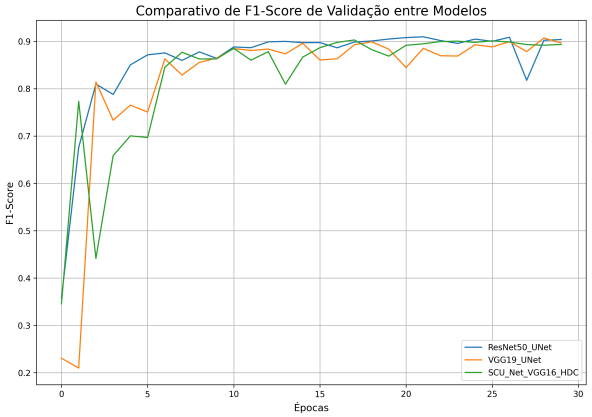


Figura 7 – Gráfico comparativo relacionado a métrica *F1-score* durante a etapa de validação para cada modelo. Fonte: Próprio Autor.

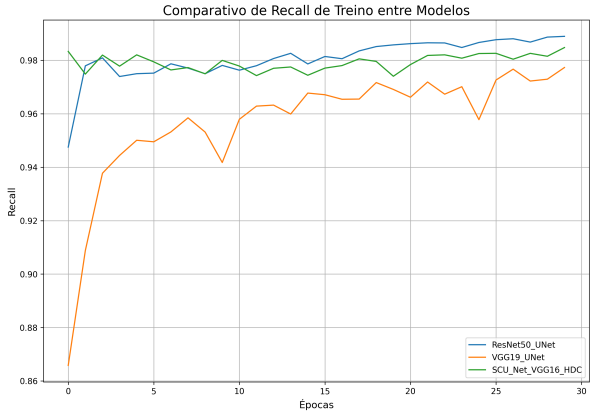


Figura 8 – Gráfico comparativo relacionado a métrica *Recall* durante a etapa de treino para cada modelo. Fonte: Próprio Autor.

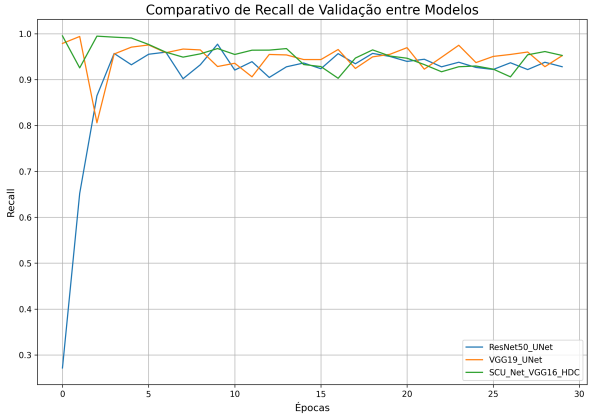


Figura 9 – Gráfico comparativo relacionado a métrica *Recall* durante a etapa de validação para cada modelo. Fonte: Próprio Autor.

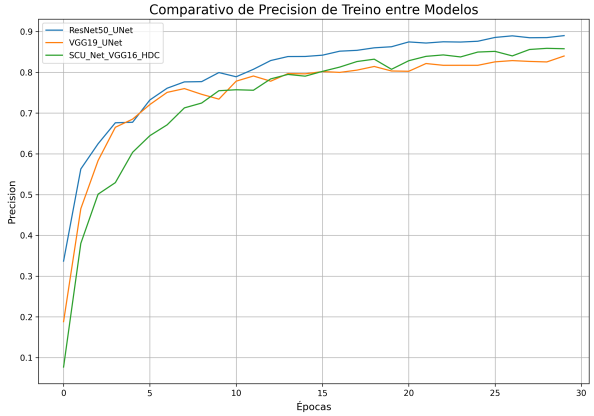


Figura 10 – Gráfico comparativo relacionado a métrica *Precision* durante a etapa de treino para cada modelo. Fonte: Próprio Autor.

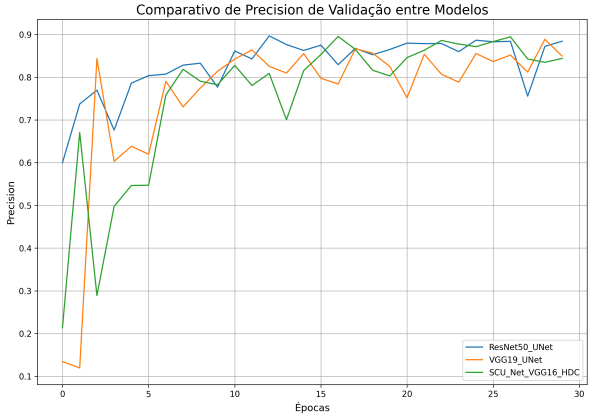


Figura 11 – Gráfico comparativo relacionado a métrica *Precision* durante a etapa de validação para cada modelo. Fonte: Próprio Autor.

Tabela 1 – Tabela comparativa entre os modelos em relação as melhores métricas na etapa de validação *Weight Tversky Loss*, *F1-score* e *Dice coefficient*. Fonte: Próprio Autor.

Modelo	**Val weig. Loss**	**Val F1-Score**	**Val Dice Coeff**
ResNet50_UNet	0.148798	0.895574	0.894956
VGG19_UNet	0.151119	0.894823	0.888155
SCU-Net_VGG16_HDC	0.179182	0.877726	0.855743

Tabela 2 – Tabela comparativa entre os modelos em relação as melhores métricas na etapa de validação *Precision* e *Recall*. Fonte: Próprio Autor.

Modelo	**Val Precision**	**Val Recall**
ResNet50_UNet	0.841761	0.959662
VGG19_UNet	0.838773	0.961096
SCU-Net_VGG16_HDC	0.816566	0.953774

4.1.2 Análise qualitativa e visual do desempenho dos modelos

Os resultados qualitativos são apresentados nas figuras 12 e figura 13 abaixo:

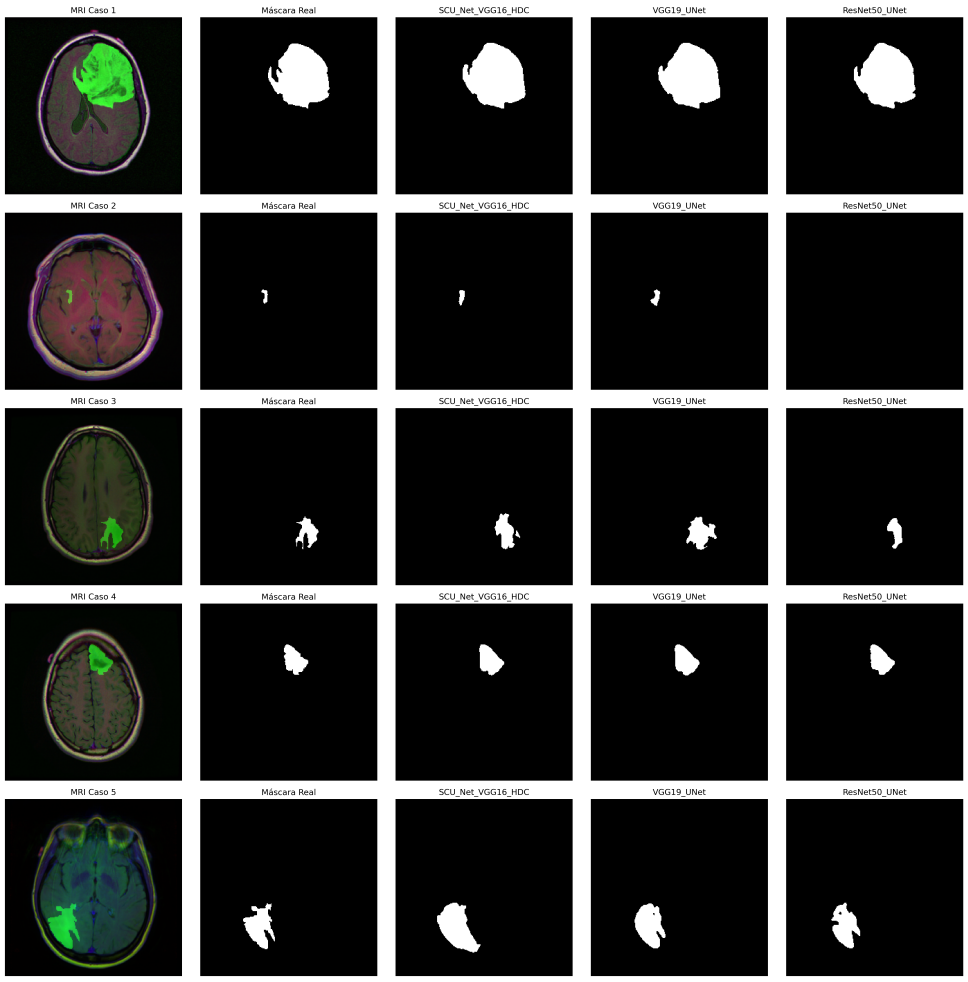
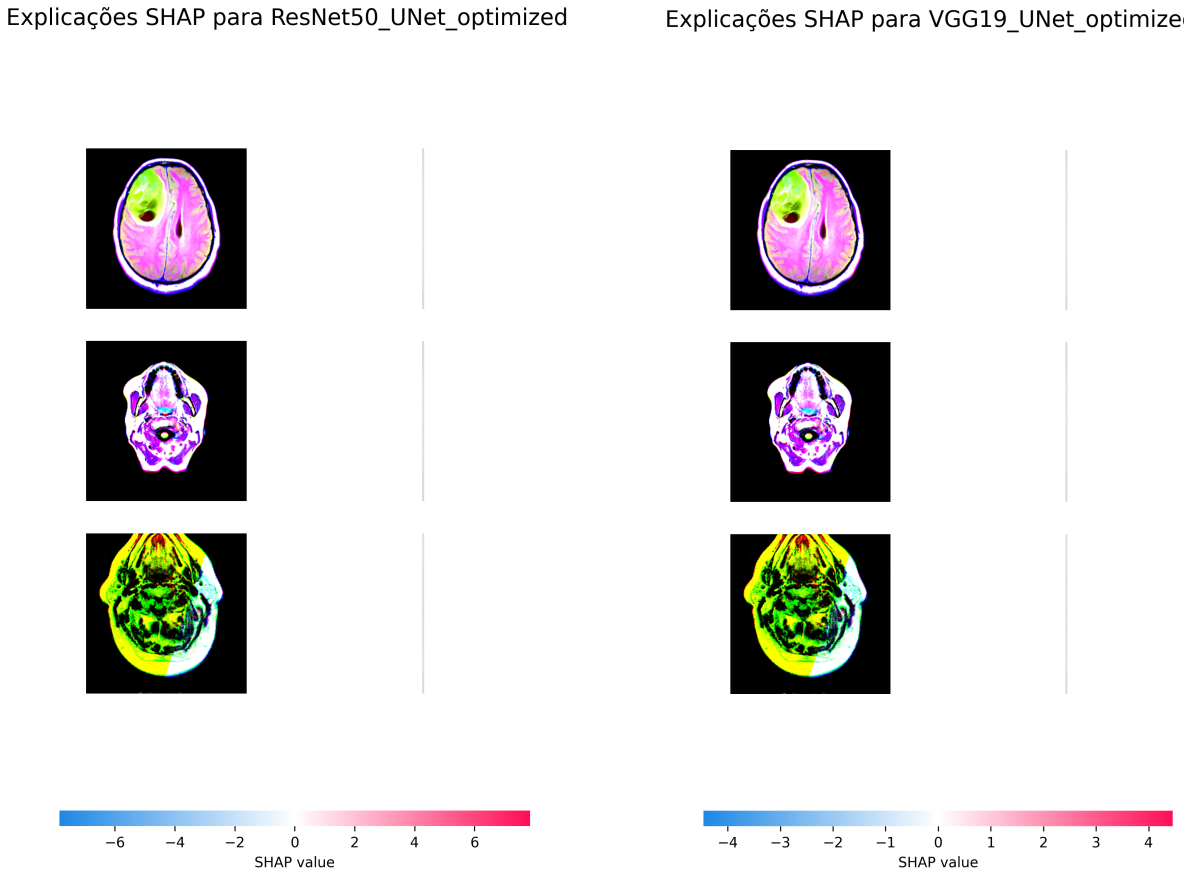
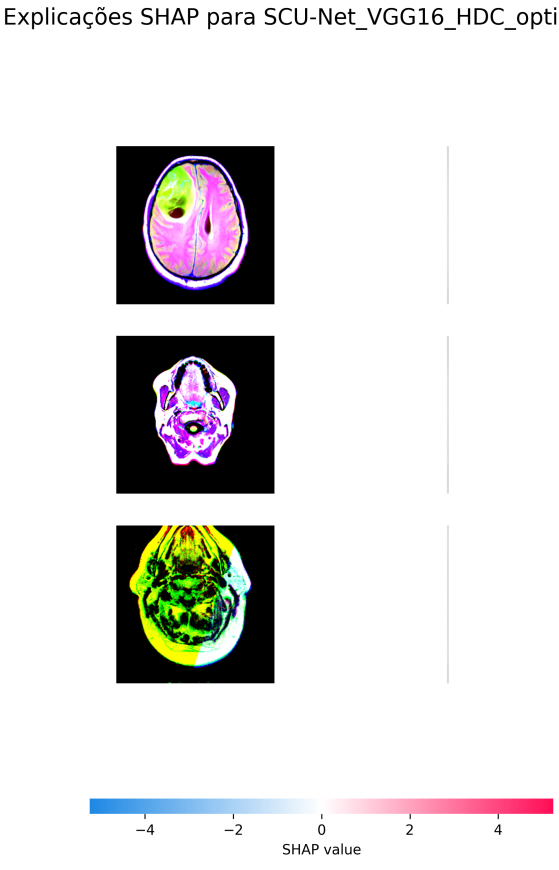


Figura 12 – Imagem comparativa entre as máscaras reais segmentadas e as máscaras previstas por cada modelo treinado. Fonte: Próprio Autor.



(a) (b)



(c)

Figura 13 – Imagens explicativas dos modelos com uso da ferramenta *SHAP*: (a) *ResNet50_UNet*, (b) *VGG19-UNet* e (c) *SCU-Net_VGG16_HDC*. Fonte: Próprio Autor.

4.2 Discussão

A análise das métricas de validação mostrou que o modelo *ResNet50_UNet* obteve um desempenho global levemente superior, alcançando um *Dice Coefficient* médio de 0.894956, seguido pelos modelos *VGG19-UNet* que apresentou um valor de 0.888155 e *SCU-Net_VGG16_HDC* com um valor de 0.855743, de acordo com a tabela 1. Esse resultado corrobora a hipótese de que a profundidade e a estrutura residual da *ResNet50_UNet* favorecem a extração de características mais discriminativas, especialmente em regiões de borda, cruciais para a segmentação médica.

A métrica de perda ponderada *Weighted Tversky Loss* apresentou um menor valor para o modelo *ResNet50_UNet* de 0.148798, se comparado com os modelos *VGG19-UNet* e *SCU-Net_VGG16_HDC*, apresentando valores de 0.151119 e 0.179182, respectivamente, de acordo com a tabela 1, indicando menor penalização por falsos negativos, aspecto importante em diagnósticos clínicos, onde a não detecção de uma região tumoral pode ser mais crítica que a inclusão de falsos positivos. Os valores de *Precision* e *Recall* também reforçam esse equilíbrio: O modelo *ResNet50_UNet* obteve um valor de *Recall* de 0.959662, de acordo com a tabela 2, evidenciando boa sensibilidade, enquanto o modelo *SCU-Net* mostrou leve tendência a subsegmentar regiões marginais da lesão.

A análise das explicações visuais geradas com *SHAP*, apresentadas na figura 13, revelou padrões consistentes com as diferenças de desempenho entre as arquiteturas. Nos mapas *SHAP* do *ResNet50_UNet*, observam-se regiões de maior contribuição concentradas nas bordas das lesões, representadas por áreas de coloração quente (vermelho/laranja), ou seja, indica que o modelo baseia suas decisões nas transições de intensidade entre tecido saudável e patológico, comportamento desejável para segmentação médica.

Por outro lado, o modelo *VGG19-UNet* apresentou regiões de relevância mais difusas, cobrindo amplamente a área tumoral, o que sugere que o modelo captura contextos mais globais, mas com menor precisão nos contornos. Já o modelo *SCU-Net_VGG16_HDC* mostrou maior foco em áreas internas homogêneas da lesão, mas pouca ênfase nos limites externos, refletindo diretamente sua menor pontuação em métricas de contorno.

Esses resultados indicam que a abordagem *SHAP* não apenas confirma o desempenho quantitativo, mas também valida o comportamento anatômico esperado dos modelos. As regiões destacadas coincidem com áreas muito intensas nas imagens *FLAIR*, reforçando a consistência clínica das inferências realizadas.

A etapa de otimização de hiperparâmetros com *OPTUNA* revelou diferenças de custo computacional relevantes entre as arquiteturas. O modelo *ResNet50_UNet*, por possuir mais parâmetros e blocos residuais, demandou maior tempo de treinamento por época, embora tenha convergido com menos flutuações de perda e aprendizado mais estável. O modelo *VGG19-UNet* apresentou treinamento mais rápido, porém com variação mais perceptível entre épocas. Em contrapartida, o modelo *SCU-Net_VGG16_HDC*, apesar de incorporar convoluções dilatadas que aumentam o campo receptivo, exigiu mais memória e apresentou lentidão moderada na retropropagação.

Os gráficos comparativos apresentadas nas figuras 2 à 11, apresentam evolução consistente das métricas de treino e validação, com convergência suave das curvas do *Dice* e *F1-score* nos três modelos. Essa estabilidade sugere que o processo de regularização via *Early Stopping* e *ReduceLROnPlateau* foi eficaz, evitando *overfitting*.

A interpretação conjunta das figuras e das imagens de predição, apresentadas na fígura 12, confirma que o modelo *ResNet50_UNet* produziu máscaras mais precisas, com contornos bem definidos e reduzido vazamento em áreas não tumorais. Em contrapartida,

o modelo *VGG19-UNet* apresentou sobreposição leve em regiões adjacentes e o *SCU-Net* evidenciou menor sensibilidade em áreas de borda.

Essa integração entre análise visual e quantitativa ilustra a relação direta entre as métricas de acurácia e o comportamento explicativo observado com SHAP, fortalecendo a credibilidade dos resultados.

De forma geral, os resultados foram bem próximos e não apresentam uma diferença muito significativa, ou seja, os modelos *ResNet50_UNet, VGG19-UNet e SCU-Net_VGG16_HDC* são igualmente viáveis para a implementação em uma rotina clínica de diagnóstico de tumores encefálicos. A escolha do modelo de segmentação na rotina clínica se daria a partir de fatores como: facilidade de implementação do modelo, o tempo de treinamento, esforço computacional e recursos disponíveis.

5 CONCLUSÃO

Este trabalho teve como objetivo principal avaliar o desempenho comparativo de diferentes arquiteturas de redes neurais convolucionais aplicadas à segmentação de tumores encefálicos, bem como investigar a interpretabilidade das decisões desses modelos por meio da ferramenta *SHAP*. Essa abordagem permitiu uma análise ampla, fundamental para a viabilidade de aplicação clínica no contexto de diagnósticos.

Os resultados obtidos indicaram que o modelo *ResNet50_UNet* apresentou um desempenho global levemente superior em relação aos outros modelos, alcançando valores mais elevados de *Dice Coefficient* e *F1-score*, além de melhor equilíbrio entre *Precision* e *Recall*. Esse desempenho superior foi associado à capacidade do modelo *ResNet50_UNet* de extrair e combinar informações por meio de seus blocos residuais, o que favorece a identificação precisa das bordas tumorais e a redução de falsos negativos. O modelo *VGG19-UNet*, embora também tenha apresentado resultados satisfatórios, evidenciou maior dispersão nos mapas de relevância e leve tendência à supersegmentação, enquanto o *SCU-Net_VGG16_HDC* mostrou limitações na definição de contornos, devido à sua menor profundidade e foco em regiões internas homogêneas.

A utilização da função de perda *Weighted Focal Tversky Loss* mostrou-se eficiente para lidar com o desbalanceamento das classes e reduzir o impacto de falsos negativos, proporcionando uma maior estabilidade durante o treinamento. Já a aplicação de uma função de otimização via *OPTUNA*, contribuiu significativamente para a seleção automatizada de hiperparâmetros, garantindo melhor convergência e reprodutibilidade dos resultados.

As análises realizadas com a ferramenta *SHAP* possibilitaram compreender de forma interpretável as regiões da imagem que mais influenciaram as predições dos modelos, evidenciando coerência entre as regiões de maior importância e as áreas clinicamente relevantes das lesões tumorais. A integração entre análise quantitativa e qualitativa mostrou que a interpretabilidade, amplia a confiança no uso de modelos de *Deep Learning* em contextos médicos.

Como perspectivas futuras, pensamos em expandir o estudo incorporando métricas complementares e utilizar-se de técnicas de explicabilidade adicionais, como *Grad-CAM*, assim como realizar um *Benchmark* comparativo sistemático, visando aprimorar a confiabilidade e a aplicabilidade clínica dos sistemas de apoio ao diagnóstico.

Referências[1]

ABRAHAM, N.; KHAN, N. M. A novel focal tversky loss function with improved attention u-net for lesion segmentation. In: IEEE. *2019 IEEE 16th international symposium on biomedical imaging (ISBI 2019)*. [S.l.], 2019. p. 683–687. Citado na página 23.

AHMED, S. F. et al. Deep learning modelling techniques: current progress, applications, advantages, and challenges. *Artificial Intelligence Review*, Springer, v. 56, n. 11, p. 13521–13617, 2023. Citado na página 22.

ASIRI, A. A. et al. *Enhancing brain tumor diagnosis: an optimized CNN hyperparameter model for improved accuracy and reliability. PeerJ Computer Science 10 (2024), e1878.* 2024. Citado na página 12.

AYADI, W. et al. Deep cnn for brain tumor classification. *Neural processing letters*, Springer, v. 53, n. 1, p. 671–700, 2021. Citado na página 11.

BUDA, M. Lgg mri segmentation. *Kaggle*, Kaggle, 2019. Disponível em: ⟨https://www.kaggle.com/datasets/mateuszbuda/lgg-mri-segmentation⟩. Citado na página 16.

CHEN, H.; LUNDBERG, S.; LEE, S.-I. Explaining models by propagating shapley values of local components. In: *Explainable AI in Healthcare and Medicine: Building a Culture of Transparency and Accountability*. [S.l.]: Springer, 2020. p. 261–270. Citado na página 25.

CHOWDHURY, P.; SRIVASTAVA, G. Enhanced classification and segmentation of brain tumors in mri images using custom cnn and u-net models with xai. In: SPRINGER. *International Conference on Pattern Recognition*. [S.l.], 2024. p. 1–16. Citado na página 23.

EBRAHIMI, M. S.; ABADI, H. K. Study of residual networks for image recognition. *arXiv e-prints*, p. arXiv–1805, 2018. Citado na página 20.

HE, K. et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: *Proceedings of the IEEE international conference on computer vision*. [S.l.: s.n.], 2015. p. 1026–1034. Citado na página 20.

HE, K. et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 770–778. Citado na página 12.

HE, K. et al. Identity mappings in deep residual networks. In: SPRINGER. *European conference on computer vision*. [S.l.], 2016. p. 630–645. Citado na página 20.

HWANG, J.-S. et al. Determination of optimal batch size of deep learning models with time series data. *Sustainability*, MDPI, v. 16, n. 14, p. 5936, 2024. Citado na página 22.

IOFFE, S.; SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: PMLR. *International conference on machine learning*. [S.l.], 2015. p. 448–456. Citado 2 vezes nas páginas 20 e 21.

ISLAM, M. A.; JIA, S.; BRUCE, N. D. How much position information do convolutional neural networks encode? *arXiv preprint arXiv:2001.08248*, 2020. Citado na página 17.

[1] De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

KANDEL, I.; CASTELLI, M. The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. *ICT express*, Elsevier, v. 6, n. 4, p. 312–315, 2020. Citado na página 22.

LUNDBERG, S. M.; LEE, S.-I. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, v. 30, 2017. Citado 2 vezes nas páginas 24 e 25.

MEZZAH, S.; TARI, A. Practical hyperparameters tuning of convolutional neural networks for eeg emotional features classification. *Intelligent Systems with Applications*, Elsevier, v. 18, p. 200212, 2023. Citado na página 21.

MOSTAFIZ, R. et al. Mri-based brain tumor detection using the fusion of histogram oriented gradients and neural features. *Evolutionary Intelligence*, Springer, v. 14, n. 2, p. 1075–1087, 2021. Citado 2 vezes nas páginas 7 e 38.

MUNOZ, M. et al. The national cancer registry of uruguay: A model for sustainable cancer registration in latin-america: 718. *Asia-Pacific Journal of Clinical Oncology*, v. 10, p. 110, 2014. Citado na página 11.

NOEL, M. M. et al. Biologically inspired oscillating activation functions can bridge the performance gap between biological and artificial neurons. *Expert Systems with Applications*, Elsevier, v. 266, p. 126036, 2025. Citado na página 22.

Núcleo de Computação Científica (NCC), UNESP. *Infraestrutura — Documentação GridUnesp*. 2022. Acessado em: dd mês ano. Disponível em: ⟨https://www.ncc.unesp.br/gridunesp/docs/v2/infraestrutura.html⟩. Citado na página 21.

PAMUNGKAS, Y. et al. Impact of hyperparameter tuning on resnet-unet models for enhanced brain tumor segmentation in mri scans. *International Journal of Robotics & Control Systems*, v. 5, n. 2, 2025. Citado 3 vezes nas páginas 12, 21 e 22.

PATRO, B.; YADAV, U.; YADAV, N. Brain tumor segmentation using unet with vgg19. In: IEEE. *2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT)*. [S.l.], 2024. v. 1, p. 1–6. Citado 3 vezes nas páginas 11, 18 e 19.

PAVIGLIANITI, A. et al. A comparison of deep learning techniques for arterial blood pressure prediction. *Cognitive Computation*, Springer, v. 14, n. 5, p. 1689–1710, 2022. Citado na página 21.

POURMAHBOUBI, A.; SAEED, N. A.; TABRIZCHI, H. A brain tumor segmentation enhancement in mri images using u-net and transfer learning. *BMC Medical Imaging*, Springer, v. 25, n. 1, p. 307, 2025. Citado na página 19.

RAIAAN, M. A. K. et al. A systematic review of hyperparameter optimization techniques in convolutional neural networks. *Decision Analytics Journal*, Elsevier, v. 11, p. 100470, 2024. Citado na página 21.

SALIH, A. M. et al. A perspective on explainable artificial intelligence methods: Shap and lime. *Advanced Intelligent Systems*, Wiley Online Library, v. 7, n. 1, p. 2400304, 2025. Citado na página 24.

Secretaria de Saúde do Distrito Federal. Câncer de cérebro: dados apontam mais de 11 mil novos casos ao ano. *Portal da Secretaria de Saúde do Distrito Federal*, Governo do Distrito Federal, maio 2024. Publicado em 06/05/2024, atualizado em 17/07/2024 às 13h56. Acesso em: 15 set. 2025. Disponível em: ⟨https://www.saude.df.gov.br/web/guest/w/c%C3%A2ncer-de-c%C3%A9rebro-dados-apontam-mais-de-11-mil-novos-casos-ao-ano⟩. Citado na página 11.

SHAO, Y. et al. An improved bge-adam optimization algorithm based on entropy weighting and adaptive gradient strategy. *Symmetry*, MDPI, v. 16, n. 5, p. 623, 2024. Citado na página 22.

SHARIF, M. I. et al. M3btcnet: multi model brain tumor classification using metaheuristic deep neural network features optimization. *Neural Computing and Applications*, Springer, v. 36, n. 1, p. 95–110, 2024. Citado na página 11.

SHEHAB, L. H. et al. An efficient brain tumor image segmentation based on deep residual networks (resnets). *Journal of King Saud University-Engineering Sciences*, Elsevier, v. 33, n. 6, p. 404–412, 2021. Citado 5 vezes nas páginas 7, 20, 21, 38 e 39.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. Citado na página 12.

TAVARES, A. R. et al. Integração de redes neurais e inteligência artificial no diagnóstico de tumores cerebrais: convergência entre tecnologia e saúde. *Cuadernos de Educación y Desarrollo*, v. 16, n. 13, p. e7065–e7065, 2024. Citado na página 13.

TEMME, M. Algorithms and transparency in view of the new general data protection regulation. *Eur. Data Prot. L. Rev.*, HeinOnline, v. 3, p. 473, 2017. Citado na página 13.

WANG, P. et al. Understanding convolution for semantic segmentation. In: IEEE. *2018 IEEE winter conference on applications of computer vision (WACV)*. [S.l.], 2018. p. 1451–1460. Citado na página 17.

XU, C.; COEN-PIRANI, P.; JIANG, X. Empirical study of overfitting in deep learning for predicting breast cancer metastasis. *Cancers*, MDPI, v. 15, n. 7, p. 1969, 2023. Citado na página 22.

YANG, T. et al. Improving brain tumor segmentation on mri based on the deep u-net and residual units. *Journal of X-ray Science and Technology*, SAGE Publications Sage UK: London, England, v. 28, n. 1, p. 95–110, 2020. Citado 2 vezes nas páginas 11 e 12.

ZEINELDIN, R. A. et al. Explainability of deep neural networks for mri analysis of brain tumors. *International journal of computer assisted radiology and surgery*, Springer, v. 17, n. 9, p. 1673–1683, 2022. Citado na página 13.

ZHAO, X. et al. A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*, Springer, v. 57, n. 4, p. 99, 2024. Citado na página 22.

ZHENG, P.; ZHU, X.; GUO, W. Brain tumour segmentation based on an improved u-net. *BMC Medical Imaging*, Springer, v. 22, n. 1, p. 199, 2022. Citado 6 vezes nas páginas 7, 11, 12, 17, 18 e 37.

Apêndice A – Arquiteturas Utilizadas nos Modelos

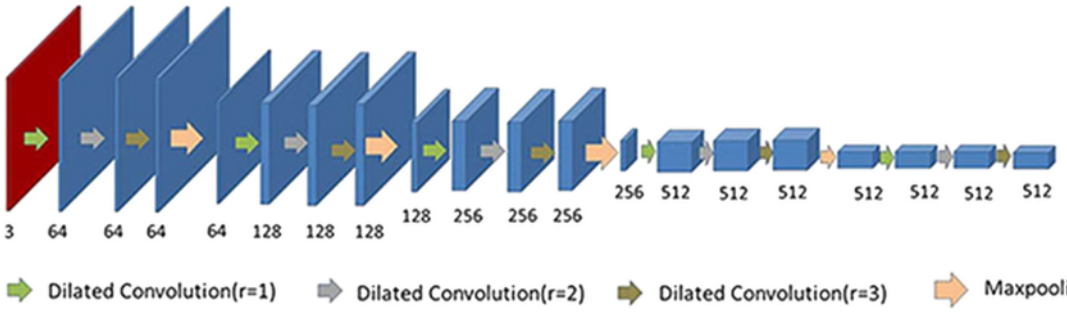


Figura 14 – Estrutura da rede de codificação do modelo SCU-Net_VGG16_HDC com convoluções dilatadas. Fonte:(ZHENG; ZHU; GUO, 2022).

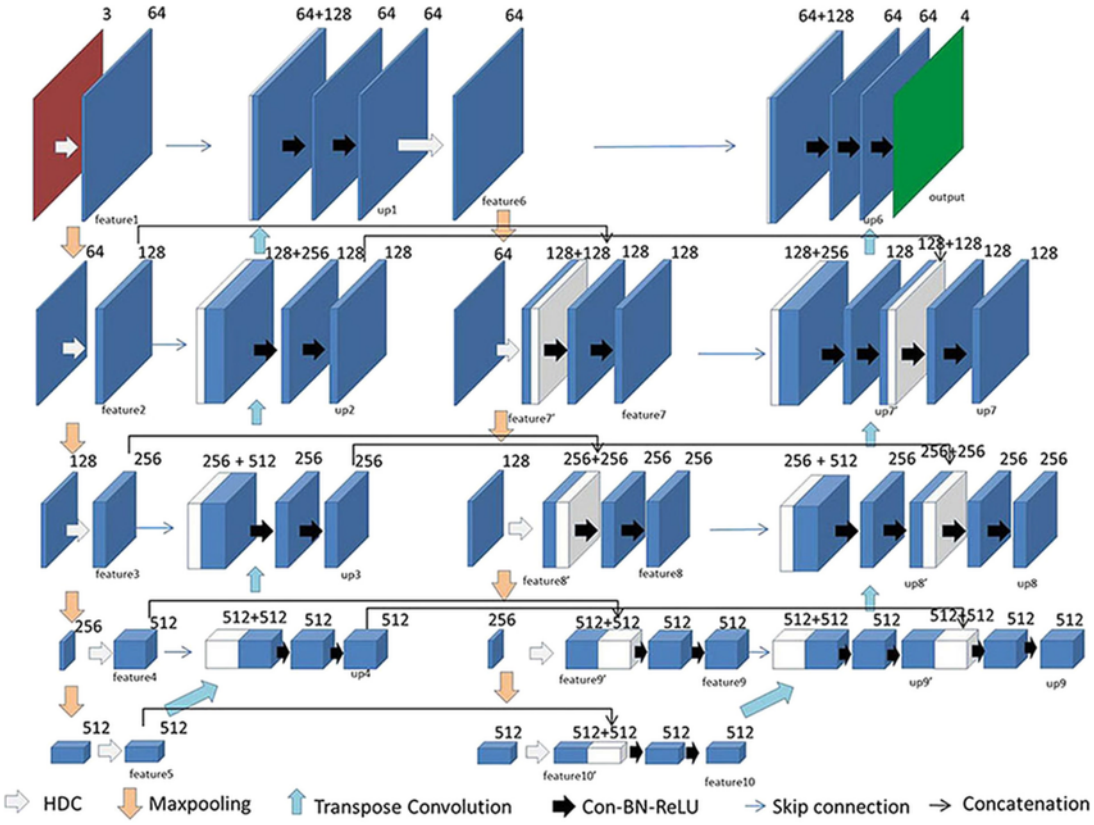


Figura 15 – Arquitetura geral do modelo SCU-Net_VGG16_HDC. Fonte:(ZHENG; ZHU; GUO, 2022).

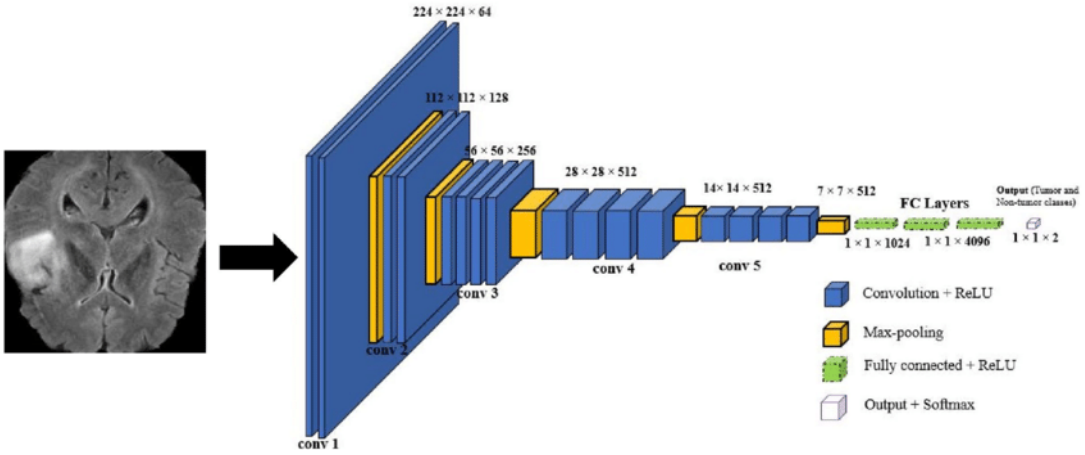


Figura 16 – Arquitetura da parte codificadora VGG19. Fonte:(MOSTAFIZ et al., 2021).

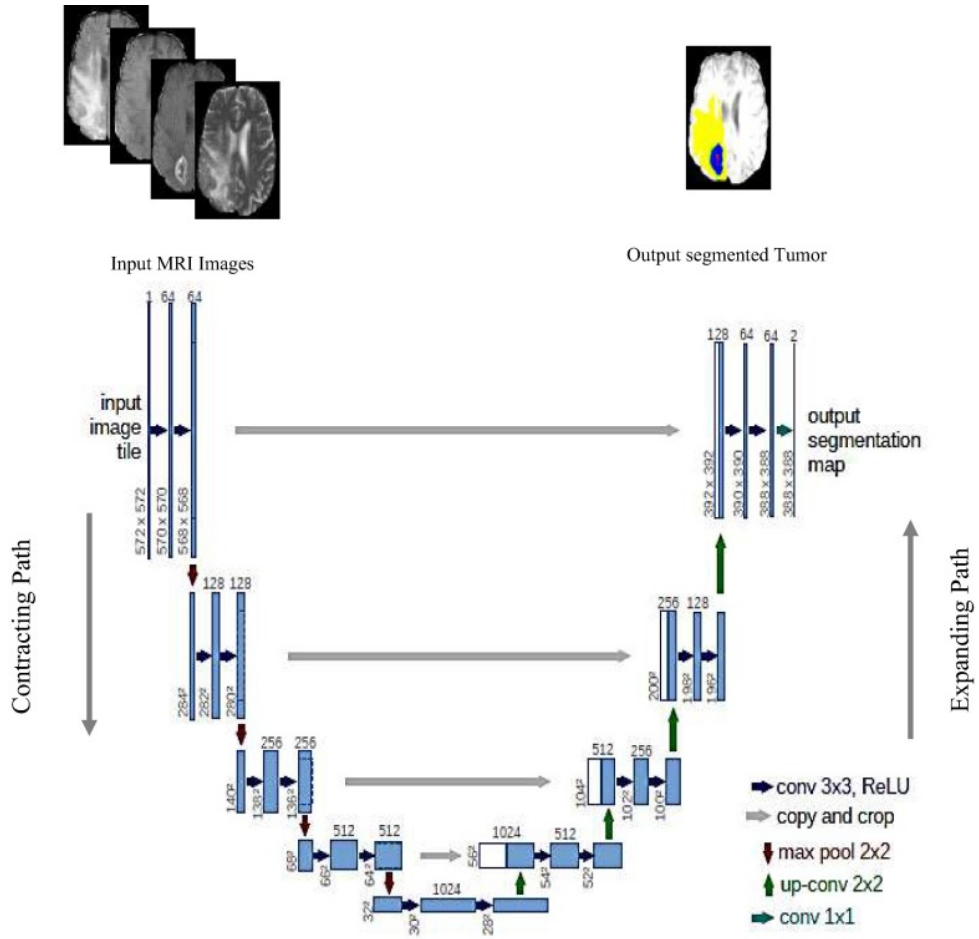


Figura 17 – Representação da arquitetura U-Net. Fonte:(SHEHAB et al., 2021).

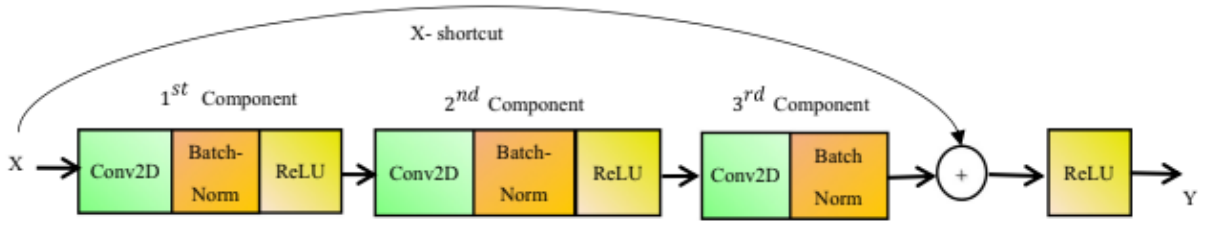


Figura 18 – Estrutura do bloco de identidade e seus componentes. Fonte:(SHEHAB et al., 2021).

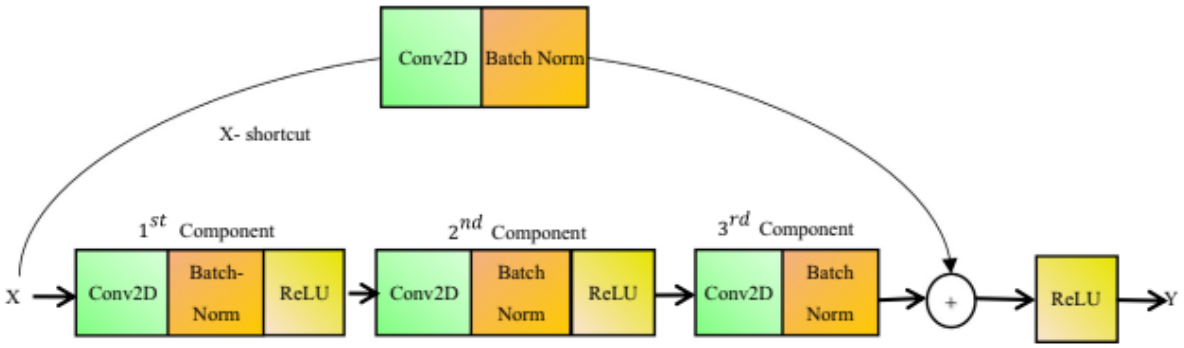


Figura 19 – Estrutura do bloco convolucional do modelo ResNet. Fonte:(SHEHAB et al., 2021).

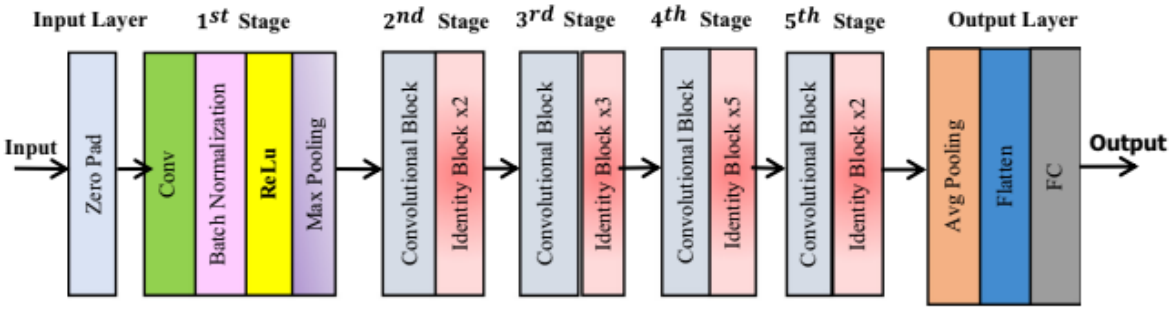


Figura 20 – Arquitetura do modelo ResNet50_UNet com blocos de identidade e convolução. Fonte:(SHEHAB et al., 2021).

Apêndice B – Algoritmos utilizados

Algoritmo 1 Função de Seleção de Hiperparâmetros *objective()*

Require: Objeto *trial*, *model_builder*, *model_name*, dados X_{train}, X_{val}, *input_shape*, pesos $weight_{pos}$ e $weight_{neg}$, número de épocas *epochs*

Ensure: Melhor valor de métricas obtidas na validação

Limpar sessão do Keras
Imprimir início do *trial* com o nome do modelo
Sortear hiperparâmetros:
    $learning\_rate \leftarrow trial.suggest\_float(10^{-5}, 10^{-2}, log = True)$
    $epsilon \leftarrow trial.suggest\_float(10^{-9}, 10^{-6}, log = True)$
    $patience\_lr \leftarrow trial.suggest\_int(3, 8)$
    $patience\_es \leftarrow trial.suggest\_int(7, 15)$
    $batch\_size \leftarrow trial.suggest\_categorical([4, 8, 16])$
Criar geradores de treino e validação com *batch_size*
Construir o modelo $model \leftarrow model\_builder(input\_shape)$
Definir otimizador Adam com $(learning\_rate, \epsilon)$
Definir função de perda *Weighted Focal Tversky Loss*
Compilar modelo com métricas: *Tversky, Dice, Precision, Recall, F1*
Criar lista de *callbacks*:
    EarlyStopping monitorando *val_loss*
    ReduceLROnPlateau com fator 0.2 e $\Delta_{min} = 10^{-4}$
Treinar modelo com *epochs*, conjunto de validação e *callbacks*
Armazenar histórico $hist \leftarrow history.history$
function GET_BEST_METRIC($history, metric\_name$)
    $key \leftarrow$ *val_* $+ metric\_name$
    if *key* existe em *history* **then**
        return $\max(history[key])$
    else
        return 0.0
Calcular melhores métricas:
    $best\_f1 \leftarrow get\_best\_metric(hist, "f1\_metric")$
    $best\_dice \leftarrow get\_best\_metric(hist, "dice\_coef")$
    $best\_tversky \leftarrow get\_best\_metric(hist, "tversky\_basic")$
    $best\_precision \leftarrow get\_best\_metric(hist, "precision\_metric")$
    $best\_recall \leftarrow get\_best\_metric(hist, "recall\_metric")$
Imprimir métricas de validação formatadas
return $best\_f1$, $best\_dice$, $best\_tversky$, $best\_precision$, $best\_recall$

Algoritmo 2 Função de Execução da Otimização de Hiperparâmetros `run_hyperparameter_tuning()`

Require: `model_builder`, `model_name`, dados `X_train_df`, `X_val_df`, `input_shape`, `weight_pos`, `weight_neg`, `n_trials`, `epochs_per_trial`

Ensure: Melhores hiperparâmetros encontrados no processo de otimização

1: Imprimir início da otimização com `model_name`
2: Definir função `objective_with_args` que invoca a função `objective` (Algoritmo 1) com os argumentos fixos
3: `study` ← Criar estudo de otimização para maximização
4: Executar otimização do `study` por `n_trials`
5: `best_trial` ← Obter o melhor trial do `study`
6: Imprimir o valor da métrica de validação do `best_trial`
7: Imprimir os hiperparâmetros (‘chave: valor‘) do `best_trial`
8: **return** os hiperparâmetros do `best_trial`