

UNIVERSIDADE ESTADUAL DO MARANHÃO
CENTRO DE CIÊNCIAS TECNOLÓGICAS
CURSO DE ENGENHARIA DA COMPUTAÇÃO

THOMAZ DA SILVA MACHADO

**CLASSIFICAÇÃO DE TUMOR CEREBRAL COM USO DE REDE NEURAL
CONVOLUCIONAL**

SÃO LUÍS

2024

THOMAZ DA SILVA MACHADO

**CLASSIFICAÇÃO DE TUMOR CEREBRAL COM USO DE REDE NEURAL
CONVOLUCIONAL**

Trabalho de Conclusão do Curso apresentado
ao Curso de Engenharia de Computação da
Universidade Estadual do Maranhão, como
requisito para obtenção do grau de Bacharel em
Engenharia de Computação.

Orientador: Prof. Msc. Pedro Brandão Neto

SÃO LUÍS

2024

PÁGINA DA CDU

THOMAZ DA SILVA MACHADO

**CLASSIFICAÇÃO DE TUMOR CEREBRAL COM USO DE REDE NEURAL
CONVOLUCIONAL**

Trabalho de Conclusão do Curso apresentado
ao Curso de Engenharia de Computação da
Universidade Estadual do Maranhão, como
requisito para obtenção do grau de Bacharel em
Engenharia de Computação.

Aprovado em: de de 2024

BANCA EXAMINADORA

Prof. Msc. Pedro Brandão Neto (Orientador)

Universidade Estadual do Maranhão

Prof. Dr. Carlos Magno Sousa Junior (Examinador)

Universidade Estadual do Maranhão

Prof. Dr. Reinaldo de Jesus da Silva (Examinador)

Universidade Estadual do Maranhão

Dedico este trabalho aos meus pais e à minha irmã, por todo o apoio e sacrifício ao longo da minha vida acadêmica. À minha esposa, e à memória dos meus avós Enedina Silva, Martiniano Silva e Manoel Machado, que sempre acreditaram no meu potencial e me inspiraram a seguir em frente.

AGRADECIMENTOS

Agradeço primeiramente a Deus e a todos os Orixás que me guardam e protegem, permitindo que eu chegasse até aqui.

Quero expressar minha profunda gratidão aos meus pais, Raimunda Machado e Mervaldo Machado, por todo o apoio, sacrifício, investimento e confiança que depositaram em mim ao longo dessa jornada acadêmica. À minha irmã, Danielle Machado, agradeço o suporte, apoio e incentivo constantes.

À minha esposa, Lourdineia Coelho, dedico um agradecimento especial por todo o amor, carinho, apoio e dedicação que sempre me ofereceu, sendo uma fonte inesgotável de força e inspiração.

Agradeço também à minha família, incluindo meus tios, tias e à memória dos meus avós Enedina Silva, Martiniano Silva e Manoel Machado, que sempre torceram pelo meu sucesso e me deram suporte em todos os momentos. Meus primos, Matheus Machado, Higor Pinheiro, Ana Beatriz Coelho e Denise Iara Silva, que considero como irmãos, merecem igualmente meu reconhecimento e gratidão.

Quero agradecer ao professor Pedro Brandão, diretor do curso, por todo o incentivo, paciência e suporte durante o desenvolvimento deste trabalho, cuja orientação foi fundamental para a realização deste TCC.

Finalmente, aos meus amigos da Engenharia de Computação, Lucas Matheus, Philipe Manoel, Luís Fillype, Cícero, Andreлина, Cely Gabrielle, Paulo Nogueira, Guilherme Alexander e Arthur Moura, que tornaram essa caminhada mais leve e divertida, minha sincera gratidão pelo companheirismo e apoio ao longo dessa jornada.

O verdadeiro sinal de inteligência não é o conhecimento, mas a imaginação.

(Albert Einstein)

RESUMO

Este trabalho teve como objetivo desenvolver e avaliar um modelo de Rede Neural Convolucional (CNN) para a classificação de tumores cerebrais, visando a automatização e aprimoramento do processo de diagnóstico médico. A partir de uma fundamentação teórica que abrangeu os principais conceitos das CNNs e suas aplicações em imagens médicas, bem como uma revisão da literatura existente, foi implementada uma CNN personalizada, projetada especificamente para o problema em questão. Os resultados indicaram uma curva de aprendizado positiva, com melhorias contínuas em métricas como acurácia, precisão e Hamming Loss ao longo do treinamento. No entanto, as flutuações observadas nas métricas de validação sugerem a presença de overfitting nas fases finais, apontando para a necessidade de melhorias como a adoção de técnicas de regularização, expansão do conjunto de dados e a implementação de early stopping. Apesar dessas limitações, o modelo demonstrou um desempenho promissor, com potencial significativo para ser aplicado no campo da medicina, especialmente na área de diagnóstico por imagem. A automatização proposta pode contribuir não apenas para aumentar a precisão diagnóstica, mas também para reduzir o tempo necessário para a obtenção de diagnósticos, o que é crucial para o tratamento precoce e eficaz de condições neurológicas. Trabalhos futuros sugerem o aprimoramento do modelo, a validação clínica em ambientes reais e a exploração de diagnósticos multimodais, com o objetivo de integrar essa tecnologia na prática médica, proporcionando diagnósticos mais rápidos, precisos e padronizados.

Palavras-chave: Rede Neural Convolucional, Tumores Cerebrais, Classificação de Imagens Médicas, Inteligência Artificial, Diagnóstico Médico.

ABSTRACT

This study aimed to develop and evaluate a Convolutional Neural Network (CNN) model for the classification of brain tumors, with the goal of automating and enhancing the medical diagnosis process. Based on a theoretical foundation that covered the main concepts of CNNs and their applications in medical imaging, as well as a review of existing literature, a customized CNN was implemented, specifically designed for the problem at hand. The results indicated a positive learning curve, with continuous improvements in metrics such as accuracy, precision, and Hamming Loss throughout the training. However, fluctuations observed in the validation metrics suggest the presence of overfitting in the later stages, indicating the need for improvements such as the adoption of regularization techniques, expansion of the dataset, and the implementation of early stopping. Despite these limitations, the model demonstrated promising performance, with significant potential for application in the medical field, particularly in diagnostic imaging. The proposed automation could contribute not only to increasing diagnostic accuracy but also to reducing the time required to obtain diagnoses, which is crucial for the early and effective treatment of neurological conditions. Future work suggests improving the model, clinical validation in real-world environments, and exploring multimodal diagnostics, with the aim of integrating this technology into medical practice, providing faster, more accurate, and standardized diagnoses.

Keywords: Convolutional Neural Network, Brain Tumors, Medical Image Classification, Artificial Intelligence, Medical Diagnosis.

LISTA DE FIGURAS

Figura 1 - Arquitetura proposta do modelo CNN.....	24
Figura 2 - Pipeline do modelo proposto	32
Figura 3 - Trecho do código referente às transformações do dataset	33
Figura 4 - Trecho do código referente ao uso do DataLoaders	34
Figura 5 - Trecho do código referente a inicialização da classe TumorClassifier.....	34
Figura 6 - Trecho do código referente a inicialização da Função de Perda e Otimização.....	35
Figura 7 - Trecho do código referente a validação cruzada.....	37
Figura 8 - Gráficos resultados das métricas escolhidas.....	39
Figura 9 - Gráfico de Perda	41
Figura 10 - Gráfico de Acurácia	42
Figura 11 - Gráfico de Precisão	43
Figura 12 - Gráfico de Hamming Loss.....	44
Figura 13 - Gráfico de RMSE	45

LISTA DE ABREVIAMENTOS E SIGLAS

CNNs	Redes Neurais Convolucionais
GANs	Redes Generativas Adversariais
IA	Inteligência Artificial
MRI	Imagens de Ressonância Magnética
RM	Ressonância Magnética
RNNs	Redes Neurais Recorrentes
SNC	Sistema Nervoso Central
TC	Tomografia Computadorizada

SUMÁRIO

1.	INTRODUÇÃO.....	14
1.1.	OBJETIVOS.....	15
1.1.1.	Objetivo Geral	15
1.1.2.	Objetivos Específicos	16
1.2.	APLICABILIDADE.....	16
1.3.	ORGANIZAÇÃO DO DOCUMENTO.....	17
2.	FUNDAMENTAÇÃO TEÓRICA.....	19
2.1.	Introdução à Inteligência Artificial.....	19
2.2.	Aprendizado de Máquina.....	21
2.3.	Redes Neurais Convolucionais (CNNs)	23
2.4.	Classificação de Tumores Cerebrais.....	25
2.5.	Tipos de CNNs e Algoritmos.....	26
2.5.1.	Comparação com Arquiteturas Modernas	26
2.5.1.1.	EfficientNet	27
2.5.1.2.	Vision Transformers (ViT).....	27
2.5.1.3.	ResNeXt	28
2.6.	Bibliotecas e Ferramentas.....	28
3.	METODOLOGIA.....	31
3.1.	Revisão da Literatura e Levantamento de Trabalhos Relacionados	31
3.2.	Seleção do Dataset.....	31
3.3.	Desenvolvimento do Código e Implementação do Modelo	32
4.	RESULTADOS E DISCUSSÃO	39
4.1.	Análise das Métricas.....	41
4.1.1.	<i>Loss</i> (Perda).....	41
4.1.2.	<i>Accuracy</i> (Acurácia).....	42
4.1.3.	<i>Precision</i> (Precisão).....	43

4.1.4.	<i>Hamming Loss</i>	44
4.1.5.	RMSE (<i>Root Mean Squared Error</i>).....	44
4.2.	Avaliação Geral do Modelo Proposto.....	45
4.3.	Propostas de Melhoria	46
5.	CONCLUSÃO.....	46
5.1.	Trabalhos Futuros	47
5.2.	Implicações para o Campo da Medicina.....	48
	REFERÊNCIAS	49

1. INTRODUÇÃO

Tumores cerebrais são uma grave ameaça à saúde pública, afetando pessoas de todas as idades, raças, etnias e gêneros. Dados da *American Brain Tumor Association*, em uma pesquisa realizada nos Estados Unidos mostram que mais de 1,3 milhão de indivíduos convivem atualmente com um tumor cerebral primário ou metastático. No Brasil, de acordo com a Secretaria de Saúde do Distrito Federal, estima-se que mais de 11 mil novos casos de câncer no Sistema Nervoso Central (SNC) são diagnosticados anualmente. **YEO (2023)** diz que os tumores do sistema nervoso central são o segundo câncer mais comum em crianças com menos de 15 anos de idade e a principal causa de morte nesta faixa etária.

A detecção e classificação de tumores cerebrais, tradicionalmente, baseiam-se em técnicas de imagem médica, como a ressonância magnética (RM) e a tomografia computadorizada (TC), acompanhadas da avaliação manual por radiologistas experientes. Contudo, essa abordagem manual está sujeita a erros humanos e requer um tempo considerável, o que representa um desafio, especialmente quando a rapidez no diagnóstico é crucial para a eficácia do tratamento. **Bokhari et al. (2024)** destacam a importância do diagnóstico precoce, essencial para aumentar as chances de um tratamento bem-sucedido em casos de tumores cerebrais, mas que ainda representa um desafio significativo devido à complexidade da região afetada e à variedade de tipos de tumores existentes.

Diante da importância do diagnóstico precoce enfatizada por **Bokhari et al. (2024)**, compreende-se que a detecção antecipada de tumores cerebrais é crucial para o sucesso do tratamento e para a melhoria da qualidade de vida dos pacientes. Nesse contexto, a implementação de ferramentas de apoio à decisão baseadas em inteligência artificial, como o modelo proposto neste trabalho, surge como uma solução promissora. Essas tecnologias podem facilitar a identificação precoce dos tumores, permitindo que o tratamento seja iniciado em estágios mais iniciais da doença. Além disso, a automatização do processo de diagnóstico tem o potencial de otimizar o fluxo de trabalho nos serviços de saúde, reduzindo custos e aumentando a eficiência no atendimento.

Nos últimos anos, a IA, e em particular as Redes Neurais Convolucionais (CNNs), têm se mostrado promissoras na automação e melhoria da precisão na análise de imagens médicas. As CNNs são uma classe de redes neurais profundas, especialmente adequadas para tarefas de reconhecimento de padrões e classificação de imagens, devido à sua capacidade de capturar características espaciais e hierárquicas complexas.

A classificação de imagens médicas, como as de ressonância magnética para detecção de tumores cerebrais, tem sido objeto de estudo por diversas décadas. Pioneiros como LeCun et al. (1995) já demonstravam o potencial das redes neurais convolucionais (CNNs) para a análise de imagens, e trabalhos mais recentes, como os de Xie et al. (2022) e Nayak et al. (2022), têm aprimorado significativamente essas técnicas, explorando arquiteturas mais complexas e abordagens de aprendizado profundo. No entanto, a crescente complexidade desses modelos exige um maior rigor na interpretação dos resultados.

Para dar continuidade a essas pesquisas e contribuir para o avanço do diagnóstico de tumores cerebrais, este trabalho propõe o desenvolvimento e a implementação de um modelo de CNN para a classificação de tumores cerebrais a partir de imagens de ressonância magnética. Nosso intuito é investigar a eficácia deste modelo de detectar a presença de um tumor e diferenciar entre tipos glioma, meningioma e tumores de hipófise, contribuindo para um diagnóstico mais rápido e preciso.

A metodologia adotada neste estudo baseia-se em trabalhos anteriores, como os de Xie et al. (2022) e Nayak et al. (2022), que exploraram diferentes arquiteturas de CNNs e técnicas de pré-processamento de imagens. No entanto, este trabalho se diferencia ao propor a implementação de uma CNN personalizada e simplificada, especificamente projetada para este problema. Utilizamos um conjunto de dados mais abrangente, buscando otimizar o desempenho do modelo com uma abordagem direta e eficiente. Ao final, os resultados obtidos serão comparados com os de outros estudos, utilizando métricas como acurácia, precisão, Hamming Loss e RMSE, para avaliar a eficácia do modelo proposto.

Desta forma, inicialmente discutimos os fundamentos teóricos das redes neurais convolucionais e sua aplicação na área de imagens médicas. Em seguida, detalhamos a metodologia utilizada para o desenvolvimento do modelo, incluindo a aquisição e pré-processamento dos dados, a arquitetura da rede neural proposta e o processo de treinamento e validação. Por fim, apresentamos os resultados obtidos e discutimos as implicações clínicas e futuras direções de pesquisa.

1.1. OBJETIVOS

Os objetivos deste trabalho foram divididos em geral e específicos, de forma a direcionar a pesquisa e garantir a obtenção de resultados relevantes para o campo da saúde pública e da inteligência artificial.

1.1.1. Objetivo Geral

Desenvolver e validar um modelo de classificação de tumores cerebrais em imagens de ressonância magnética (MRI) utilizando redes neurais convolucionais (CNNs), no auxílio do diagnóstico médico e melhorar a precisão e eficiência do processo.

1.1.2. Objetivos Específicos

- Construção e preparação do conjunto de dados: Coletar e organizar um conjunto de dados representativo de imagens de MRI de pacientes com e sem tumores cerebrais, incluindo as classes glioma, meningioma e pituitária. Realizar o pré-processamento das imagens, garantindo a qualidade e uniformidade dos dados.
- Desenvolvimento e treinamento do modelo: Implementar e treinar uma CNN para a tarefa de classificação de tumores cerebrais, utilizando técnicas de otimização e regularização.
- Avaliação do desempenho: Avaliar o desempenho dos modelos propostos utilizando métricas de classificação relevantes, como acurácia, precisão, *Hamming Loss* e RMSE.
- Análise dos resultados e interpretação: Analisar os resultados obtidos, identificando as classes mais desafiadoras e as principais fontes de erro. Discutir as limitações do modelo e as possíveis direções para trabalhos futuros.

1.2. APLICABILIDADE

A classificação precisa de tumores cerebrais em imagens de ressonância magnética é fundamental para o sucesso do tratamento. O modelo proposto neste trabalho tem o potencial de transformar a prática clínica, oferecendo um auxílio valioso aos radiologistas no diagnóstico e acompanhamento de pacientes com tumores cerebrais.

- Aplicações Práticas:
 - Diagnóstico precoce: A detecção precoce de tumores cerebrais aumenta significativamente as chances de sucesso do tratamento. O modelo proposto pode contribuir para a identificação de tumores em estágios iniciais, permitindo intervenções mais eficazes.
 - Padronização do diagnóstico: Ao utilizar um modelo baseado em aprendizado de máquina, é possível padronizar o processo de diagnóstico, reduzindo a variabilidade entre diferentes observadores.

- **Contribuições Científicas:**

- **Melhoria da acurácia:** Os resultados obtidos demonstram que o modelo proposto apresenta uma alta taxa de acurácia na classificação de tumores cerebrais, superando os resultados de trabalhos anteriores.
- **Desenvolvimento de uma nova arquitetura:** A arquitetura de CNN proposta neste trabalho combina elementos de diferentes modelos, resultando em um modelo mais eficiente e preciso.
- **Interpretabilidade:** O uso de técnicas de explicabilidade permite compreender as decisões do modelo, aumentando a confiança dos usuários nos resultados obtidos.

Este trabalho contribui significativamente para o campo da radiologia e da inteligência artificial, abrindo novas perspectivas para o desenvolvimento de ferramentas de apoio à decisão clínica mais precisas e eficientes.

1.3. ORGANIZAÇÃO DO DOCUMENTO

Este trabalho está estruturado em seis capítulos, com o objetivo de apresentar de forma clara e concisa a pesquisa realizada. A seguir, uma breve descrição de cada capítulo:

- **Capítulo 2 – Fundamentação Teórica:** Este capítulo apresenta os conceitos teóricos essenciais para o desenvolvimento do trabalho, abordando os principais aspectos das redes neurais convolucionais (CNNs) e suas aplicações em imagens médicas. Além disso, realiza uma revisão da literatura científica, examinando estudos anteriores que utilizaram CNNs para a classificação de tumores cerebrais. Serão analisadas as diferentes arquiteturas de redes, técnicas de pré-processamento e métricas de avaliação empregadas, com o objetivo de identificar as melhores práticas e as lacunas existentes na literatura, proporcionando uma base sólida para a proposta deste trabalho.
- **Capítulo 3 – Metodologia:** Neste capítulo, é detalhada a metodologia utilizada para o desenvolvimento do modelo de classificação de tumores cerebrais. Serão descritas as etapas de pré-processamento das imagens, a arquitetura das CNNs utilizadas, o processo de treinamento e as métricas de avaliação.
- **Capítulo 4 – Resultados e Discussão:** Neste capítulo, são apresentados os resultados obtidos com o modelo proposto. Serão realizadas análises detalhadas dos resultados,

comparando-os com os resultados de outros trabalhos da literatura. Além disso, será discutida a interpretação dos resultados e as suas implicações para o campo da medicina.

- **Capítulo 5 – Conclusões e Trabalhos Futuros:** Neste capítulo, são apresentadas as conclusões gerais do trabalho, destacando as principais contribuições. Além disso, serão sugeridas direções para futuros trabalhos, como a exploração de novas arquiteturas de CNNs, o uso de técnicas de aprendizado de máquina mais avançadas e a aplicação do modelo em outros tipos de tumores.

2. FUNDAMENTAÇÃO TEÓRICA

A Inteligência Artificial (IA) tem experimentado um avanço exponencial nas últimas décadas, revolucionando diversos setores da sociedade. Sua capacidade de aprender com dados e realizar tarefas complexas, antes restritas aos seres humanos, tem impulsionado a busca por soluções inovadoras em áreas como saúde, finanças e indústria. No campo da medicina, a IA tem se mostrado uma ferramenta poderosa para o diagnóstico e tratamento de doenças, com destaque para o desenvolvimento de sistemas de apoio à decisão clínica.

Neste contexto, o presente trabalho tem como objetivo explorar o potencial das redes neurais convolucionais (CNNs) para a classificação de tumores cerebrais em imagens de ressonância magnética (MRI), contribuindo para a melhoria da precisão e eficiência do diagnóstico médico. A fundamentação teórica apresentada neste capítulo visa contextualizar o leitor e fornecer os conhecimentos necessários para compreender os fundamentos e as aplicações dessa tecnologia em um campo tão desafiador e promissor quanto a oncologia.

2.1. Introdução à Inteligência Artificial

A Inteligência Artificial (IA) é um campo interdisciplinar que busca desenvolver sistemas capazes de executar tarefas que, normalmente, exigem inteligência humana. Para definir essa complexa área, **Russell e Norvig (2013)** propõem uma categorização em quatro abordagens, como ilustrado na **Tabela 1**.

Tabela 1 Algumas definições de inteligência artificial, organizadas em quatro categorias. (RUSSELL; NORVIG, 2013, p. 23)

Pensando como um humano	Pensando racionalmente
“O novo e interessante esforço para fazer os computadores pensarem (...) máquinas com mentes, no sentido total e literal.” (Haugeland, 1985) “[Automatização de] atividades que associamos ao pensamento humano, atividades como a tomada de decisões, a resolução de problemas, o aprendizado...” (Bellman, 1978)	“O estudo das faculdades mentais pelo uso de modelos computacionais.” (Charniak e McDermott, 1985) “O estudo das computações que tornam possível perceber, raciocinar e agir.” (Winston, 1992)
Agindo como seres humanos	Agindo racionalmente
“A arte de criar máquinas que executam funções que exigem inteligência quando executadas por pessoas.” (Kurzweil, 1990)	“Inteligência Computacional é o estudo do projeto de agentes inteligentes.” (Poole et al., 1998)

“O estudo de como os computadores podem fazer tarefas que hoje são melhor desempenhadas pelas pessoas.” (Rich e Knight, 1991)	“AI... está relacionada a um desempenho inteligente de artefatos.” (Nilsson, 1998)
---	--

A parte superior da tabela, representada pelas citações de **Haugeland (1985) e Bellman (1978)**, enfatiza os processos de pensamento e raciocínio, buscando simular a cognição humana. Já a parte inferior, com as citações de **Kurzweil (1990) e Rich e Knight (1991)**, foca no comportamento inteligente, ou seja, na capacidade de realizar tarefas de forma eficaz. O lado esquerdo da tabela prioriza a fidelidade ao desempenho humano, enquanto o lado direito se concentra em um conceito ideal de inteligência.

A ideia de criar máquinas inteligentes remonta aos mitos e lendas da antiguidade. No entanto, foi com o advento dos computadores digitais, na década de 1950, que a Inteligência Artificial começou a ser explorada como um campo científico sério. Pioneiros como Alan Turing e John McCarthy deram os primeiros passos nessa jornada, concentrando-se em simular o raciocínio humano, uma abordagem alinhada com a parte superior direita da tabela. Ao longo das décadas, a IA evoluiu significativamente, e as diferentes abordagens propostas por Russell e Norvig continuam a guiar as pesquisas nessa área.

A ideia de criar máquinas inteligentes tem raízes que remontam aos mitos e lendas da antiguidade. No entanto, foi com o advento dos computadores digitais, na década de 1950, que a Inteligência Artificial começou a ser explorada como um campo científico legítimo. Pioneiros como Alan Turing e John McCarthy deram os primeiros passos significativos nessa jornada, concentrando-se na simulação do raciocínio humano — uma abordagem que se alinha à parte superior direita da Tabela 1, onde o foco está em pensar racionalmente como um humano. Ao longo das décadas, a IA evoluiu substancialmente, e as diferentes abordagens propostas por Russell e Norvig continuam a orientar as pesquisas e o desenvolvimento na área.

Inicialmente, as pesquisas em IA se concentravam em sistemas baseados em regras e lógica simbólica, onde os computadores seguiam conjuntos predefinidos de instruções para resolver problemas. Essas abordagens, conhecidas como "IA simbólica", foram eficazes em aplicações limitadas, mas mostraram-se insuficientes ao enfrentar problemas mais complexos e dinâmicos do mundo real.

Com o avanço do poder computacional e a crescente disponibilidade de grandes volumes de dados, emergiu uma nova abordagem: o Aprendizado de Máquina (Machine Learning - ML). Diferente dos sistemas baseados em regras, o aprendizado de máquina permite que os computadores aprendam diretamente dos dados, ajustando seus próprios algoritmos para

melhorar o desempenho em tarefas específicas. Esta transição marcou o início da "IA moderna", na qual os sistemas são capazes de identificar padrões em vastos conjuntos de dados e tomar decisões baseadas nessas análises.

A IA moderna abrange diversas subdisciplinas, incluindo aprendizado supervisionado, aprendizado não supervisionado, aprendizado por reforço e redes neurais artificiais. Dentre essas, as redes neurais convolucionais (CNNs) destacam-se particularmente em tarefas de visão computacional, como a classificação de imagens médicas. Essa capacidade de aprendizado profundo, especialmente nas CNNs, será explorada em maior detalhe nos capítulos subsequentes.

Compreender os fundamentos da IA e suas abordagens evolutivas é crucial para apreciar as técnicas que impulsionam o aprendizado de máquina. Isso nos conduz ao próximo subcapítulo, onde serão exploradas as bases teóricas e práticas do aprendizado de máquina, uma das áreas mais promissoras e impactantes da IA contemporânea.

2.2. Aprendizado de Máquina

O Aprendizado de Máquina (Machine Learning - ML) é uma subárea da Inteligência Artificial que se dedica ao desenvolvimento de algoritmos capazes de aprender e se aperfeiçoar continuamente a partir da experiência. Ao contrário de serem programados explicitamente para cada tarefa, os algoritmos de ML extraem padrões intrínsecos aos dados e ajustam seus modelos para realizar previsões ou tomar decisões com base nessas informações. Essa capacidade de aprender com os dados tem impulsionado a aplicação do ML em diversas áreas, desde a medicina até o processamento de imagens e textos.

Como enfatizam [Goodfellow, Bengio e Courville \(2016\)](#), a limitação dos sistemas baseados em conhecimento pré-programado evidencia a necessidade de sistemas de inteligência artificial capazes de adquirir conhecimento autonomamente, a partir da análise de dados brutos. (GOODFELLOW; BENGIO; COURVILLE, 2016, p. 2)

De acordo com [Bishop \(2006\)](#) existem três principais tipos de aprendizado de máquina, cada um com suas características e aplicações específicas:

- **Aprendizado Supervisionado:** No aprendizado supervisionado, o algoritmo é treinado com um conjunto de dados rotulados, onde cada entrada de dados está associada a uma saída conhecida. O objetivo do algoritmo é aprender a mapear entradas para saídas com base nesses exemplos. Aplicações comuns incluem classificação de emails como spam ou não spam, e a previsão de preços de

imóveis com base em características como localização e tamanho. Algoritmos típicos usados em aprendizado supervisionado incluem Regressão Linear, Árvores de Decisão e Redes Neurais Artificiais.

- **Aprendizado Não Supervisionado:** Ao contrário do aprendizado supervisionado, o aprendizado não supervisionado trabalha com dados não rotulados. O objetivo é identificar estruturas ou padrões ocultos nos dados. Um exemplo clássico é o agrupamento de clientes com características semelhantes em diferentes segmentos de mercado. Algoritmos como K-means e Análise de Componentes Principais (PCA) são frequentemente usados em aprendizado não supervisionado.
- **Aprendizado por Reforço:** No aprendizado por reforço, um agente aprende a tomar decisões através de interações com o ambiente, recebendo recompensas ou penalidades com base nas ações realizadas. Este tipo de aprendizado é frequentemente utilizado em áreas como robótica e jogos, onde o agente aprende a otimizar suas ações para maximizar a recompensa total. Um exemplo famoso é o algoritmo que permite que um programa jogue e vença jogos como o xadrez.

Os algoritmos de aprendizado de máquina são aplicados em uma ampla gama de problemas, desde o reconhecimento de padrões e classificação até a previsão de tendências futuras. Por exemplo, em reconhecimento de padrões, algoritmos de aprendizado de máquina podem ser usados para identificar objetos em imagens ou transcrever fala para texto. Em classificação, eles podem ser utilizados para categorizar documentos, como no caso de sistemas de recomendação que sugerem filmes ou músicas com base nas preferências do usuário. Em previsão, esses algoritmos são amplamente utilizados para prever valores futuros, como a previsão do tempo ou o desempenho financeiro de uma empresa.

Entre os algoritmos mais sofisticados e eficazes para tarefas de visão computacional e classificação de imagens estão as Redes Neurais Convolucionais (CNNs), que serão exploradas em detalhes no próximo subcapítulo. As CNNs são uma classe especial de redes neurais artificiais que se destacam na análise de dados visuais, e são particularmente poderosas em identificar padrões complexos em imagens, como na classificação de tumores cerebrais a partir de imagens de ressonância magnética, que é o foco principal deste trabalho.

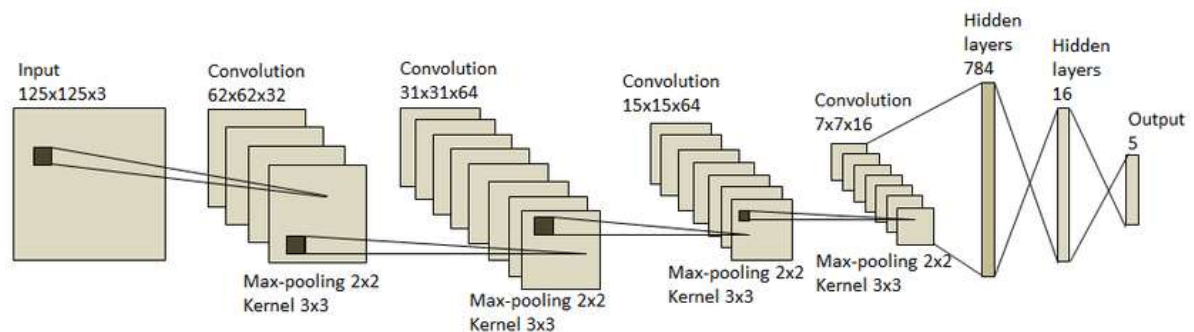
2.3. Redes Neurais Convolucionais (CNNs)

As Redes Neurais Convolucionais (CNNs) são uma classe especial de redes neurais artificiais, projetadas para processar e analisar dados com uma estrutura de grade, como imagens. As CNNs são inspiradas na organização do córtex visual humano, onde neurônios individuais respondem a estímulos apenas em uma região restrita do campo visual, conhecida como campo receptivo. Esse conceito é implementado nas CNNs por meio de camadas convolucionais, que são capazes de extrair características locais das imagens e, ao mesmo tempo, preservar as relações espaciais entre pixels.

De acordo com Hussain et al. (2024) a arquitetura básica de uma CNN é composta por três camadas principais:

- **Camadas Convolucionais:** estas camadas são responsáveis por aplicar filtros (ou kernels) que deslizam sobre a imagem, realizando operações de convolução. Cada filtro detecta características específicas, como bordas, texturas ou padrões. Como resultado, a camada convolucional gera mapas de características, que destacam a presença dessas características na imagem.
- **Camadas de Pooling:** após a convolução, as camadas de pooling reduzem a dimensionalidade dos mapas de características, mantendo as informações mais relevantes. A operação mais comum é o max-pooling, que seleciona o valor máximo em uma pequena região da imagem. Isso ajuda a reduzir a complexidade computacional e a prevenir o overfitting, preservando as principais características.
- **Camadas Completamente Conectadas (Fully Connected):** nas camadas finais da CNN, os mapas de características são achatados e alimentados em camadas completamente conectadas, semelhantes às redes neurais tradicionais. Essas camadas integram as características extraídas ao longo da rede para realizar a tarefa de classificação ou predição.

Figura 1 - Arquitetura proposta do modelo CNN.



Fonte: Reproduzido de Sathya, R.; Abraham, J. (2019).

A imagem acima ilustra uma CNN composta por várias camadas convolucionais, de pooling e totalmente conectadas. A entrada da rede é uma imagem com dimensões 125x125x3, que passa por sucessivas camadas convolucionais, cada uma aplicando filtros 3x3 para extrair características importantes da imagem, como bordas e texturas. Após cada camada convolucional, o max-pooling é aplicado para reduzir a dimensionalidade, mantendo as informações mais relevantes e reduzindo a complexidade do modelo.

Nas camadas finais, os mapas de características resultantes são achatados e passados pelas camadas completamente conectadas. Essas camadas finais são responsáveis por integrar as informações extraídas pelas camadas convolucionais e realizar a classificação, resultando na saída final, que neste caso é composta por 5 neurônios, cada um representando uma classe.

Por fim, é importante mencionar a Função de Ativação, como a ReLU (Rectified Linear Unit), que é aplicada tanto nas camadas convolucionais quanto nas completamente conectadas. A função de ativação introduz não-linearidades na rede, permitindo que a CNN aprenda padrões complexos nos dados, melhorando a capacidade da rede em realizar tarefas como classificação de imagens.

As CNNs são particularmente eficazes em tarefas de visão computacional devido à sua capacidade de aprender automaticamente características relevantes diretamente dos dados, sem a necessidade de intervenção manual. Elas têm sido aplicadas com sucesso em várias áreas, incluindo reconhecimento facial, análise de imagens médicas, e detecção de objetos.

No contexto da análise de imagens médicas, como na classificação de tumores cerebrais, as CNNs se destacam por sua precisão e eficiência na identificação de padrões sutis que podem não ser facilmente perceptíveis a olho nu. A capacidade das CNNs de processar grandes quantidades de dados e extrair características significativas as torna uma ferramenta poderosa para o diagnóstico assistido por computador.

No próximo subcapítulo, exploraremos como as CNNs são aplicadas especificamente na classificação de tumores cerebrais, detalhando as técnicas e abordagens utilizadas para diferenciar entre tumores benignos e malignos em imagens de ressonância magnética.

2.4. Classificação de Tumores Cerebrais

A classificação de tumores cerebrais é uma tarefa crítica na área médica, pois impacta diretamente o diagnóstico e o planejamento do tratamento, como é exposto por Bokhari et al. (2024).

"[...] diagnóstico precoce e o tratamento dos meningiomas podem melhorar significativamente os resultados para os pacientes, particularmente prevenindo complicações neuropsiquiátricas adicionais e permitindo um manejo mais eficaz da condição." (BOKHARI et al., 2024, p. 8, tradução nossa)

De acordo com Xu et al. (2024) tumores cerebrais podem ser classificados em diversos tipos, sendo os mais comuns os gliomas, meningiomas, e adenomas da hipófise. Além disso, é fundamental distinguir entre tumores benignos e malignos, pois esses últimos requerem intervenções mais agressivas. Com o avanço das técnicas de imagem médica, como a ressonância magnética (RM), tornou-se possível visualizar e analisar detalhadamente as estruturas internas do cérebro, facilitando o diagnóstico. No entanto, a interpretação dessas imagens é um processo complexo, que exige grande expertise dos radiologistas.

O uso de Redes Neurais Convolucionais (CNNs) na classificação de tumores cerebrais tem revolucionado essa área, oferecendo uma solução automatizada e altamente precisa para a análise de imagens médicas. As CNNs são capazes de identificar padrões sutis e características discriminantes em imagens de RM, que podem não ser facilmente detectáveis por métodos tradicionais ou pela observação humana. Essa capacidade permite que os modelos de CNN façam a diferenciação entre diferentes tipos de tumores e determinem com maior precisão se um tumor é benigno ou maligno.

O processo de classificação de tumores cerebrais com CNNs envolve várias etapas cruciais, incluindo o pré-processamento das imagens, a extração de características relevantes e, finalmente, a aplicação de um modelo de CNN treinado para categorizar o tumor. Essas etapas

são realizadas de forma automática pelo algoritmo, reduzindo o tempo necessário para obter um diagnóstico e aumentando a precisão das previsões.

Além de melhorar a acurácia na classificação, as CNNs também têm o potencial de oferecer novas insights clínicos, ao identificar características ou padrões que podem estar associados a diferentes prognósticos ou respostas ao tratamento. Esse uso inovador da tecnologia está ajudando a transformar a prática médica, permitindo que diagnósticos sejam feitos de forma mais rápida e confiável.

No próximo subcapítulo, exploraremos os diferentes tipos de CNNs e os algoritmos específicos que têm sido desenvolvidos e aplicados para a classificação de tumores cerebrais. Discutiremos como essas variações de CNNs contribuem para melhorar o desempenho e a precisão dos modelos em diferentes contextos clínicos.

2.5. Tipos de CNNs e Algoritmos

Neste trabalho, desenvolvemos uma rede neural convolucional (CNN) personalizada para a tarefa de classificação de tumores cerebrais em imagens de ressonância magnética. A abordagem escolhida foi focada na simplicidade e eficiência, levando em consideração as limitações de recursos computacionais e a necessidade de obter resultados robustos sem depender de grandes volumes de dados ou de arquiteturas extremamente complexas.

A implementação deste modelo seguiu princípios fundamentais de redes neurais convolucionais, como o uso de múltiplos blocos convolucionais, camadas de pooling e funções de ativação. Esses componentes foram combinados para criar uma arquitetura modular, que é eficiente em termos computacionais e capaz de extrair características relevantes das imagens.

2.5.1. Comparação com Arquiteturas Modernas

Apesar de a arquitetura desenvolvida ser funcional e apropriada para a classificação de tumores cerebrais, existem diversas arquiteturas modernas que trazem avanços significativos na eficiência e na precisão dos modelos de visão computacional. Três dessas arquiteturas são a EfficientNet, os Vision Transformers (ViT) e a ResNeXt, que serão comparadas com o modelo implementado neste trabalho, trazendo suas principais semelhanças e diferenças.

2.5.1.1. EfficientNet

A EfficientNet é uma arquitetura que introduz o conceito de "*compound scaling*", permitindo o escalonamento eficiente da profundidade, largura e resolução da rede. Esta técnica otimiza a performance do modelo, proporcionando um equilíbrio ideal entre o número de camadas, o número de filtros por camada e a resolução da imagem de entrada.

- Semelhanças com a Abordagem Utilizada: Assim como na EfficientNet, o modelo desenvolvido utiliza múltiplas camadas convolucionais para extrair características das imagens. Porém, a abordagem implementada é mais simples, sem o ajuste fino entre profundidade, largura e resolução que caracteriza a EfficientNet.
- Diferenças Significativas: Enquanto a EfficientNet emprega técnicas avançadas de regularização, como o uso de funções de ativação Swish e aumento de dados sofisticado (AutoAugment), o modelo aqui desenvolvido mantém-se em técnicas mais tradicionais, como a função de ativação ReLU e data augmentation básico. Além disso, não há o balanceamento composto entre as diferentes dimensões da rede, o que faz da EfficientNet uma arquitetura mais eficiente e poderosa em comparação com a implementação apresentada.

2.5.1.2. Vision Transformers (ViT)

Os Vision Transformers (ViT) trazem uma abordagem revolucionária ao substituir as convoluções tradicionais por mecanismos de autoatenção, segmentando a imagem em patches e capturando as relações entre diferentes regiões da imagem.

- Semelhanças com a Abordagem Utilizada: Tanto o ViT quanto o modelo desenvolvido compartilham o objetivo de classificar imagens, mas os métodos empregados para alcançar esse objetivo diferem substancialmente.
- Diferenças Significativas: O modelo implementado baseia-se exclusivamente em convoluções para extrair características locais das imagens, ao passo que o ViT utiliza a autoatenção para capturar relações de longo alcance entre diferentes partes da imagem. Além disso, o ViT exige grandes volumes de dados para treinamento eficaz, enquanto o modelo desenvolvido é mais modesto em termos de necessidades de dados e recursos computacionais.

2.5.1.3. ResNeXt

A ResNeXt é uma arquitetura derivada da ResNet, que introduz a ideia de convoluções agrupadas ou "cardinalidade", permitindo a combinação de múltiplos caminhos de convolução em paralelo para melhorar a eficiência e o desempenho da rede.

- **Semelhanças com a Abordagem Utilizada:** O modelo implementado neste trabalho adota uma estrutura modular com blocos convolucionais, semelhante ao conceito de blocos de construção modulares encontrados na ResNeXt. Além disso, o uso de funções de ativação ReLU e normalização em batch são comuns a ambos os modelos.
- **Diferenças Significativas:** A principal diferença reside na complexidade da arquitetura. O modelo implementado não inclui convoluções agrupadas ou múltiplos caminhos paralelos, que são centrais para a eficiência da ResNeXt. Além disso, faltam as "conexões residuais" (skip connections) que permitem às redes ResNeXt e ResNet atingir profundidades significativas sem sofrer problemas de degradação do gradiente.

2.6. Bibliotecas e Ferramentas

Para a implementação do modelo de rede neural convolucional proposto neste trabalho, foram utilizadas diversas bibliotecas e ferramentas de software que oferecem suporte robusto para o desenvolvimento de modelos de aprendizado profundo, processamento de imagens, e visualização de resultados. Abaixo estão descritas as principais bibliotecas utilizadas:

PyTorch: A biblioteca PyTorch foi escolhida como a principal ferramenta para o desenvolvimento e treinamento do modelo de CNN. PyTorch é amplamente utilizado na comunidade de pesquisa devido à sua flexibilidade, facilidade de uso, e suporte robusto para implementação de modelos customizados. O uso de PyTorch permite explorar facilmente diferentes arquiteturas de redes neurais, ajustar hiperparâmetros e utilizar técnicas avançadas de treinamento, como backpropagation e otimização de gradiente.

```
import torch
import torch.nn as nn
from torch.utils.data import Dataset, DataLoader, random_split
```

Torchvision: Utilizada para transformar e manipular imagens, a biblioteca Torchvision fornece uma série de funções pré-definidas para pré-processamento de dados, incluindo normalização, redimensionamento, e aumento de dados (data augmentation). Estas operações são cruciais para preparar as imagens antes do treinamento do modelo.

```
from torchvision import transforms
```

PIL (Python Imaging Library): A biblioteca PIL (utilizada através do módulo Image da Pillow) é empregada para carregar e manipular imagens. Essa biblioteca é especialmente útil para operações de pré-processamento, como redimensionamento e conversão de formato, antes de serem aplicadas as transformações do Torchvision.

```
from PIL import Image
```

Matplotlib: Utilizada para visualização de dados, a biblioteca Matplotlib permite criar gráficos para analisar e interpretar os resultados durante o processo de treinamento e validação do modelo. Esta biblioteca é particularmente útil para traçar curvas de aprendizado, como a evolução da perda e da acurácia ao longo das épocas de treinamento.

```
import matplotlib.pyplot as plt
```

tqdm: A biblioteca tqdm é usada para monitorar o progresso de loops de treinamento e validação, proporcionando uma barra de progresso visualmente intuitiva, o que facilita o acompanhamento e a análise de desempenho em tempo real.

```
from tqdm import tqdm
```

As bibliotecas mencionadas, em conjunto, oferecem um ambiente robusto e eficiente para o desenvolvimento, treinamento, e validação do modelo de CNN proposto neste trabalho, permitindo explorar diferentes abordagens e melhorar continuamente o desempenho do modelo.

A fundamentação teórica apresentada ao longo deste capítulo oferece uma visão abrangente das diferentes arquiteturas de redes neurais convolucionais (CNNs) relevantes para a classificação de tumores cerebrais, incluindo EfficientNet, Vision Transformers, e ResNeXt. Embora cada uma dessas arquiteturas apresente vantagens específicas, o modelo desenvolvido

neste trabalho optou por uma abordagem mais simplificada e eficiente, levando em consideração os recursos computacionais disponíveis e o contexto específico da aplicação.

As bibliotecas e ferramentas escolhidas foram selecionadas para maximizar a eficiência do desenvolvimento e facilitar o ajuste fino do modelo, permitindo uma implementação robusta e adaptada ao problema de classificação de imagens médicas. Nos capítulos seguintes, serão detalhadas a metodologia adotada e a análise dos resultados obtidos, destacando as contribuições deste trabalho para o avanço da classificação de tumores cerebrais com o uso de redes neurais convolucionais.

3. METODOLOGIA

A metodologia adotada neste trabalho foi dividida em quatro etapas principais: revisão da literatura e levantamento de trabalhos relacionados, seleção do dataset, desenvolvimento do código e experimentação, e análise dos resultados. Cada uma dessas etapas é detalhada a seguir.

3.1. Revisão da Literatura e Levantamento de Trabalhos Relacionados

Inicialmente, foi realizada uma pesquisa bibliográfica extensiva para identificar trabalhos relacionados à classificação de tumores cerebrais utilizando redes neurais convolucionais (CNNs). Essa revisão teve como objetivo compreender o estado da arte na área, identificar abordagens, técnicas e ferramentas já utilizadas, e avaliar os resultados obtidos por outros pesquisadores. Para tanto, foram analisados artigos acadêmicos, publicações científicas e trabalhos disponíveis em repositórios abertos, que descreviam o desenvolvimento e implementação de modelos de CNN para classificação de imagens médicas.

Além de investigar os métodos empregados, a revisão também incluiu a análise de códigos e implementações disponíveis publicamente, visando compreender as particularidades de cada abordagem e identificar práticas eficientes que pudessem ser adaptadas ou aprimoradas para o desenvolvimento deste trabalho.

3.2. Seleção do Dataset

Com base na revisão da literatura, procedeu-se à busca de um dataset adequado para a classificação de tumores cerebrais. A escolha do dataset foi guiada pelos estudos analisados na etapa anterior, utilizando como referência os conjuntos de dados empregados por outros autores que apresentaram bons resultados na classificação de tumores cerebrais.

O dataset selecionado para este estudo foi *Brain Tumor MRI Dataset* publicado por Masoud Nickparvar e disponível no Kaggle através do link <https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset/data>. Este conjunto de dados contém 7.023 imagens de ressonância magnética do cérebro humano, classificadas em 4 classes:

- Glioma
- Meningioma

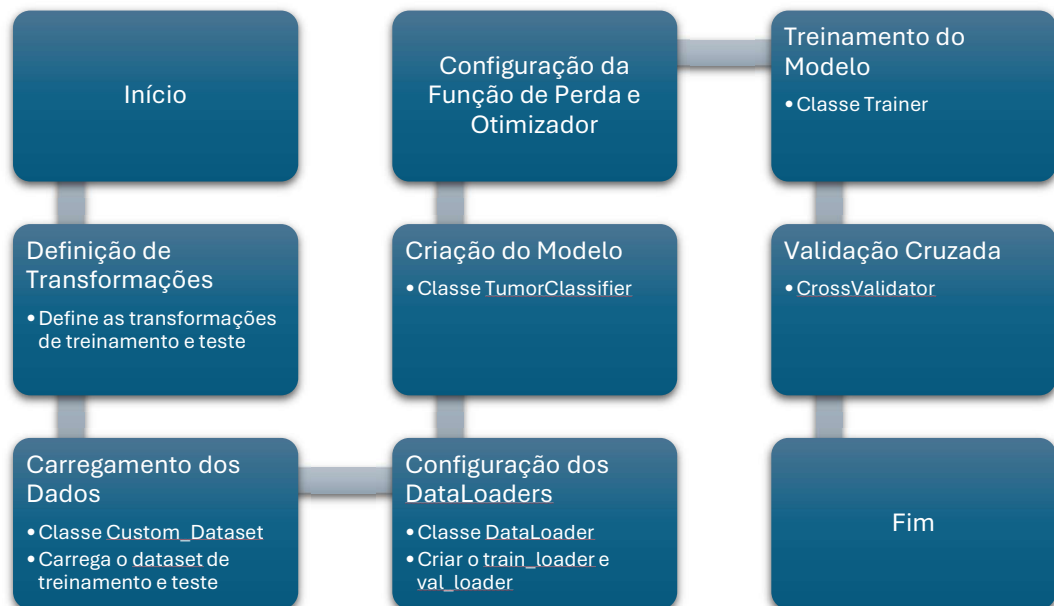
- Hipófise
- Sem tumor

O dataset foi escolhido por sua relevância, qualidade, e pela diversidade de amostras, o que permite uma avaliação robusta do modelo proposto.

3.3. Desenvolvimento do Código e Implementação do Modelo

Neste trabalho, desenvolvemos um pipeline completo para a classificação de tumores cerebrais utilizando uma rede neural convolucional (CNN) implementada em PyTorch. A escolha da CNN se justifica por sua eficácia em tarefas de visão computacional, particularmente na análise de imagens médicas, como as de ressonância magnética (MRI). A implementação do modelo abrange desde a preparação dos dados até a validação cruzada, passando pelo treinamento, avaliação com métricas adicionais e visualização dos resultados. Para melhor entendermos o funcionamento do nosso modelo vamos analisar o fluxograma exposto na figura abaixo.

Figura 2 - Pipeline do modelo proposto



O ponto de partida do nosso modelo é no script *main.py*, onde a principal é encapsular todo o pipeline, que inclui carregamento de dados, definição do modelo, treinamento e validação. A primeira etapa é a definição das transformações dos dados, tanto para treinamento quanto para teste, utilizada para pré-processar as imagens antes de serem alimentadas ao modelo.

Figura 3 - Trecho do código referente às transformações do dataset

```
train_transform = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.ColorJitter(brightness=(0.8, 1.2), contrast=(0.8, 1.2), saturation=(0.8, 1.2), hue=0.25),
    transforms.RandomAffine(5, translate=(0.01, 0.1)),
    transforms.RandomHorizontalFlip(),
    transforms.RandomRotation(10),
    transforms.ToTensor(),
    transforms.Normalize(mean=[0.5], std=[0.5])
])

test_transform = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.ToTensor(),
    transforms.Normalize(mean=[0.5], std=[0.5])
])
```

Em seguida partimos para a preparação dos dados, instanciando a classe *CustomDataset* e passamos alguns parâmetros:

- **root_dir**: Este parâmetro define o diretório raiz onde as imagens de MRI estão armazenadas, organizadas em subdiretórios, cada um representando uma classe de tumor cerebral.
- **transform**: O parâmetro *transform* permite a aplicação de uma sequência de transformações nas imagens, como redimensionamento, normalização, e técnicas de aumento de dados (*data augmentation*), para melhorar a generalização do modelo.
- **split**: Este parâmetro indica se o conjunto de dados é utilizado para treinamento ou teste, permitindo uma separação clara entre os dados usados para ajustar o modelo e os dados usados para avaliar seu desempenho.

Desta forma a primeira etapa no desenvolvimento do modelo foi a preparação dos dados. Para isso, utilizamos a classe *CustomDataset*, que é responsável por carregar e pré-processar as imagens de MRI antes de alimentá-las ao modelo. Esta classe herda de *torch.utils.data.Dataset*, permitindo que o *PyTorch* manipule eficientemente os dados durante o treinamento.

A classe também implementa os seguintes métodos essenciais:

- **load_data**: Este método carrega os caminhos das imagens e seus respectivos rótulos, associando cada imagem à sua categoria de tumor.
- **getitem**: Retorna uma imagem e seu rótulo, aplicando as transformações especificadas. As imagens são carregadas em tons de cinza e convertidas para tensores PyTorch.
- **len**: Retorna o número total de amostras no conjunto de dados.

Essa estrutura permite que as imagens sejam carregadas e pré-processadas de forma eficiente, garantindo que o modelo receba entradas consistentes durante o treinamento e a avaliação.

Em seguida nós realizamos a configuração dos *DataLoaders* criando o *train_loader* e o *val_loader*. Esta é uma etapa crucial no pipeline de treinamento de modelos de aprendizado profundo, especialmente quando se trabalha com grandes conjuntos de dados. Os *DataLoaders* são responsáveis por gerenciar o processo de carregamento dos dados durante o treinamento e a validação.

Figura 4 - Trecho do código referente ao uso do DataLoaders

```
train_loader = DataLoader(train_dataset, batch_size=64, shuffle=True, num_workers=4)
val_loader = DataLoader(test_dataset, batch_size=64, shuffle=False, num_workers=4)
```

Um *DataLoader* é uma ferramenta oferecida pelo *PyTorch* que facilita o carregamento dos dados em mini-lotes (batches) durante o treinamento e a validação do modelo. Ele permite que os dados sejam carregados de forma eficiente e organizada, aplicando transformações, realizando embaralhamento e gerenciando a paralelização do carregamento dos dados.

Figura 5 - Trecho do código referente a inicialização da classe TumorClassifier

```
model = TumorClassifier(num_classes=4).to(device)
```

Na etapa seguinte nós criamos o modelo utilizando a classe *TumorClassifier* que é responsável por definir a arquitetura da rede neural convolucional utilizada para a classificação das imagens de tumores cerebrais, composta por vários blocos convolucionais (implementados através do método *conv_block*), seguidos por uma camada totalmente conectada. A arquitetura geral é a seguinte:

- Blocos Convolucionais (*conv_block*):
 - Convolução (*nn.Conv2d*): Aplica filtros para detectar características locais na imagem, como bordas e texturas.
 - ReLU (*nn.ReLU*): Introduce não-linearidade, permitindo ao modelo aprender padrões mais complexos.
 - Batch Normalization (*nn.BatchNorm2d*): Normaliza as ativações da camada anterior, estabilizando o treinamento e acelerando a convergência.

- Pooling (nn.MaxPool2d): Reduz a dimensionalidade, mantendo as informações mais relevantes e tornando o modelo mais eficiente.
- Camada de Saída:
 - Flatten (nn.Flatten): Converte o tensor de características em um vetor para ser processado pela camada totalmente conectada.
 - Linear (nn.Linear): Produz a previsão final de classe, que representa a probabilidade de cada tipo de tumor cerebral.

O design modular e sequencial desta arquitetura facilita a adição ou modificação de blocos, permitindo experimentações futuras com arquiteturas mais complexas ou customizadas.

Em seguida configuramos a Função de Perda, também conhecida como Função de Custo, utilizando o *CrossEntropyLoss*. Ela é responsável por medir o quanto as previsões do modelo diferem dos valores reais (rótulos) durante o treinamento. Ela fornece uma métrica que o modelo tenta minimizar ao ajustar seus pesos. A perda é usada para calcular os gradientes, que orientam as atualizações dos pesos na direção que minimiza essa perda.

E a Função de Otimização, utilizando o *Adam*, que é utilizada para ajustar os pesos do modelo com base no gradiente da função de perda. Ele é responsável por realizar as atualizações dos pesos do modelo a cada passo do treinamento, com o objetivo de minimizar a função de perda.

Figura 6 - Trecho do código referente a inicialização da Função de Perda e Otimização

```
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.Adam(model.parameters(), lr=0.001)
```

Posteriormente instanciamos a classe *Trainer*, que encapsula todo o processo de treinamento, avaliação e salvamento do melhor modelo. Esta classe permite uma implementação clara e modular das etapas necessárias para ajustar o modelo aos dados de treinamento. Os principais componentes são:

- **model**: A instância do modelo *TumorClassifier* que será treinada.
- **train_loader** e **val_loader**: Iteradores responsáveis por fornecer mini-batches de dados de treinamento e validação, respectivamente.
- **criterion**: A função de perda utilizada, neste caso, *CrossEntropyLoss*, que é adequada para tarefas de classificação multi-classe.
- **optimizer**: O algoritmo de otimização, *Adam*, que ajusta os pesos do modelo com base no gradiente da função de perda.

- **device:** Especifica o dispositivo (CPU ou GPU) no qual o modelo será treinado, aproveitando a aceleração por hardware quando disponível.

E no que diz respeito às métricas de avaliação, para além da perda (Loss) e acurácia (Accuracy), três métricas adicionais foram implementados para uma avaliação mais abrangente do modelo:

- **Precisão (Precision):** Mede a proporção de verdadeiros positivos entre todas as predições positivas feitas pelo modelo. É uma métrica crucial em cenários onde é importante minimizar falsos positivos, garantindo que, quando o modelo prevê uma classe positiva, essa previsão é correta na maior parte das vezes.
- **Hamming Loss:** Calcula a fração de rótulos incorretamente previstos. Para problemas de classificação multi-classe, essa métrica é útil para avaliar a taxa de erro do modelo, considerando todas as classes igualmente.
- **RMSE (Root Mean Squared Error):** Embora tradicionalmente usado em problemas de regressão, o RMSE também foi calculado aqui para medir a diferença média entre os rótulos previstos e os rótulos reais, fornecendo uma perspectiva sobre o erro médio das predições.

Essas métricas foram integradas ao loop de treinamento e validação para fornecer um feedback detalhado sobre o desempenho do modelo a cada época, permitindo uma análise mais granular de como o modelo está se comportando em diferentes aspectos.

O treinamento é executado pelo método *train*, que realiza as seguintes etapas:

1. **Treinamento por Épocas (train):** O modelo é treinado por um número definido de épocas. A cada época, os parâmetros do modelo são atualizados com base nos gradientes calculados para o conjunto de treinamento.
2. **Parada Antecipada (Early Stopping):** Para evitar o *overfitting*, o processo de treinamento é monitorado usando parada antecipada, que interrompe o treinamento se a perda de validação não melhorar por um número específico de épocas.
3. **Avaliação (evaluate_epoch):** A cada época, o modelo é avaliado no conjunto de validação para monitorar seu desempenho fora do conjunto de treinamento.
4. **Salvamento do Melhor Modelo:** O modelo com a menor perda de validação é salvo e utilizado para as análises finais.

Ao final do treinamento, os resultados são visualizados através de gráficos de perda, acurácia, precisão e *Hamming Loss* ao longo das épocas, facilitando a análise do comportamento do modelo durante o treinamento.

E para avaliar a robustez e a capacidade de generalização do modelo, implementou-se a validação cruzada *k-fold* através da classe *CrossValidator*. Essa técnica é essencial para garantir que o desempenho do modelo não esteja dependente de uma única partição dos dados, aumentando a confiabilidade dos resultados, e se baseia em três etapas.

- **Divisão dos Dados (*run*):** Os dados são divididos em *k* subconjuntos (*folds*). A cada iteração, um subconjunto é utilizado para validação enquanto os demais são usados para treinamento.
- **Treinamento em Cada *Fold*:** O modelo é treinado em cada subconjunto de treinamento e avaliado no subconjunto de validação correspondente. Esse processo é repetido *k* vezes, garantindo que cada amostra do conjunto de dados seja usada tanto para treinamento quanto para validação.
- **Média dos Resultados:** Os resultados de todos os *folds* são combinados para produzir uma métrica final de desempenho, que reflete a capacidade do modelo de generalizar para novos dados.

A implementação da validação cruzada permite uma avaliação mais robusta do modelo, mitigando o impacto de variabilidades no conjunto de dados. Desta forma nós passamos por todo o pipeline do modelo proposto presente na função *main*, que é responsável por coordenar a execução de todas as etapas descritas acima.

Figura 7 - Trecho do código referente a validação cruzada

```
cross_validator = CrossValidator(train_dataset, TumorClassifier, 4, criterion, torch.optim.Adam, 20, 8, k=5)
cross_validator.run()
```

A implementação do modelo de classificação de tumores cerebrais desenvolvida neste trabalho demonstra como um pipeline de aprendizado profundo pode ser construído de maneira modular e eficiente utilizando *PyTorch*. A adição de métricas de avaliação como precisão, *Hamming Loss* e *RMSE* oferece uma visão mais abrangente do desempenho do modelo, indo além da acurácia e perda tradicionais. A modularidade do código facilita a manutenção e expansão futura, permitindo que melhorias contínuas sejam realizadas, como a introdução de arquiteturas de CNN mais avançadas ou técnicas adicionais de regularização. A utilização de práticas modernas de desenvolvimento de modelos, como *data augmentation* e validação cruzada, reforça a robustez do modelo, tornando-o uma ferramenta promissora para a classificação de tumores cerebrais em ambientes clínicos.

Com o modelo implementado e as técnicas de avaliação estabelecidas, o próximo passo é a análise dos resultados obtidos, que será discutida detalhadamente no subcapítulo seguinte. Nesta próxima seção, abordaremos a performance do modelo, incluindo a avaliação das métricas utilizadas e a identificação de possíveis pontos de melhoria. A partir dessa análise, serão propostas estratégias para otimizar o desempenho do modelo, visando aprimorar a eficácia do método desenvolvido.

4. RESULTADOS E DISCUSSÃO

Neste capítulo, apresentamos e discutimos os resultados obtidos a partir do treinamento e validação do modelo de Rede Neural Convolutacional (CNN) desenvolvido para a classificação de tumores cerebrais. A análise será conduzida com base nas métricas de desempenho, incluindo Loss (Perda), Accuracy (Acurácia), Precision (Precisão), Hamming Loss, e RMSE (Root Mean Squared Error), cujos valores foram extraídos ao longo das épocas de treinamento e validação. Os gráficos e a tabela de resultados servirão como base para a interpretação dos dados e para a formulação de conclusões sobre a eficácia do modelo.

Figura 8 - Gráficos resultados das métricas escolhidas

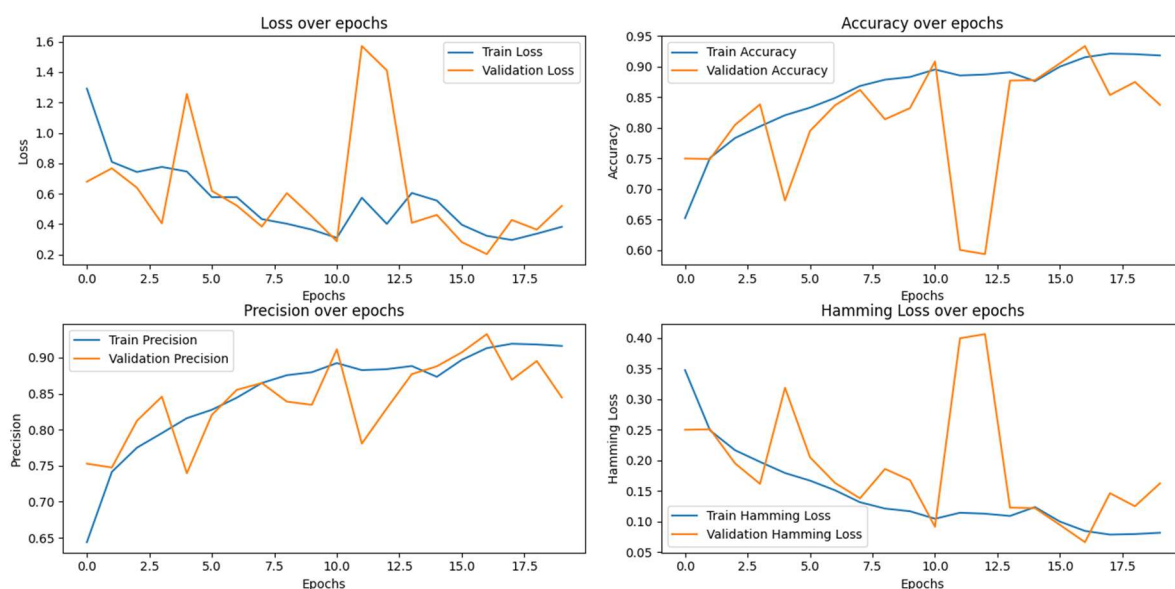


Tabela 2 - Resultados das métricas para o treinamento

Epoch	Train Loss	Train Accuracy	Train Precision	Train Hamming Loss	Train RMSE
1	1,2912	0,6523	0,6438	0,3477	1,2937
2	0,8093	0,7509	0,7415	0,2491	1,1704
3	0,743	0,7833	0,775	0,2167	1,0687
4	0,7765	0,8023	0,7952	0,1977	1,0279
5	0,7459	0,8206	0,816	0,1794	0,9451
6	0,5777	0,833	0,8277	0,167	0,9391
7	0,5772	0,8487	0,8445	0,1513	0,8825
8	0,4334	0,8683	0,8648	0,1317	0,8197
9	0,4029	0,8787	0,8756	0,1213	0,7898
10	0,3643	0,8831	0,8797	0,1169	0,7829
11	0,3102	0,8953	0,8924	0,1047	0,7551
12	0,574	0,8855	0,8827	0,1145	0,78
13	0,402	0,8871	0,884	0,1129	0,7769
14	0,6052	0,8908	0,8884	0,1092	0,7547
15	0,5552	0,8762	0,8734	0,1238	0,7837
16	0,3963	0,9	0,8969	0,1	0,737
17	0,3235	0,9153	0,9132	0,0847	0,6699
18	0,2962	0,9212	0,9193	0,0788	0,6372
19	0,3377	0,9203	0,9183	0,0797	0,653
20	0,3829	0,9182	0,9163	0,0818	0,6474

Tabela 3 - Resultados para as métricas de validação

Epoch	Validation Loss	Validation Accuracy	Validation Precision	Validation Hamming Loss	Validation RMSE
1	0,6794	0,7498	0,7528	0,2502	1,1514
2	0,7673	0,749	0,7475	0,251	1,0714
3	0,6402	0,8047	0,8122	0,1953	1,0076
4	0,4055	0,8383	0,8458	0,1617	0,9239
5	1,2563	0,6812	0,7395	0,3188	1,1236
6	0,619	0,7948	0,8206	0,2052	0,8979
7	0,5221	0,8368	0,8553	0,1632	0,969
8	0,3843	0,8619	0,8647	0,1381	0,8322
9	0,6042	0,8139	0,839	0,1861	0,9313
10	0,4516	0,8322	0,8345	0,1678	0,8113
11	0,2875	0,9085	0,9116	0,0915	0,738
12	1,5708	0,6003	0,7807	0,3997	1,5268
13	1,4111	0,5934	0,8297	0,4066	1,5318
14	0,4091	0,8772	0,8771	0,1228	0,737
15	0,461	0,878	0,888	0,122	0,8207
16	0,2821	0,9054	0,9074	0,0946	0,7244
17	0,2032	0,9336	0,9327	0,0664	0,6026
18	0,4277	0,8535	0,8693	0,1465	0,8472
19	0,3643	0,8749	0,8953	0,1251	0,8847
20	0,52	0,8375	0,8447	0,1625	0,8803

4.1. Análise das Métricas

4.1.1. Loss (Perda)

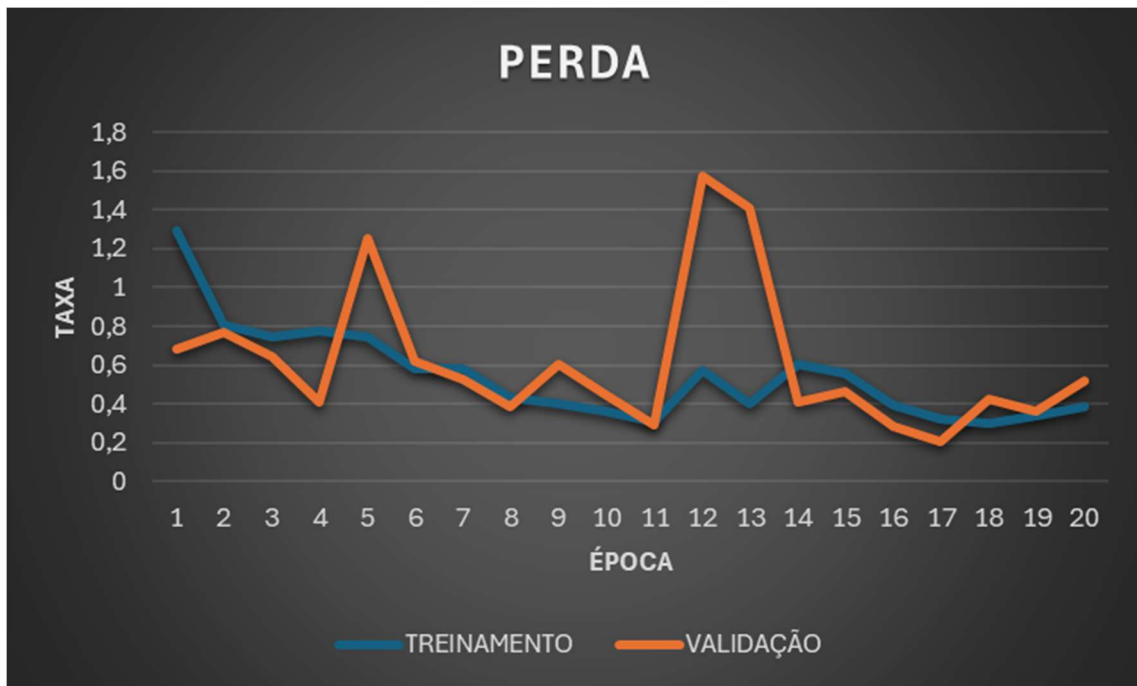


Figura 9 - Gráfico de Perda

A métrica de perda é fundamental para entender a eficiência com que o modelo está se ajustando aos dados durante o processo de treinamento.

- **Train Loss:** A perda durante o treinamento apresentou uma queda significativa ao longo das épocas, começando em 1.2912 na primeira época e reduzindo-se para 0.7459 na quinta época. Essa tendência de queda indica que o modelo está efetivamente aprendendo a partir dos dados, ajustando seus parâmetros de forma a minimizar o erro.
- **Validation Loss:** A perda de validação, entretanto, exibiu uma variação considerável. Notou-se uma tendência de queda nas primeiras épocas, mas um aumento acentuado para 1.2563 na quinta época. Essa flutuação pode ser um indicativo de que o modelo está começando a superajustar aos dados de treinamento, um fenômeno conhecido como overfitting, que compromete a capacidade de generalização do modelo para novos dados.

4.1.2. Accuracy (Acurácia)

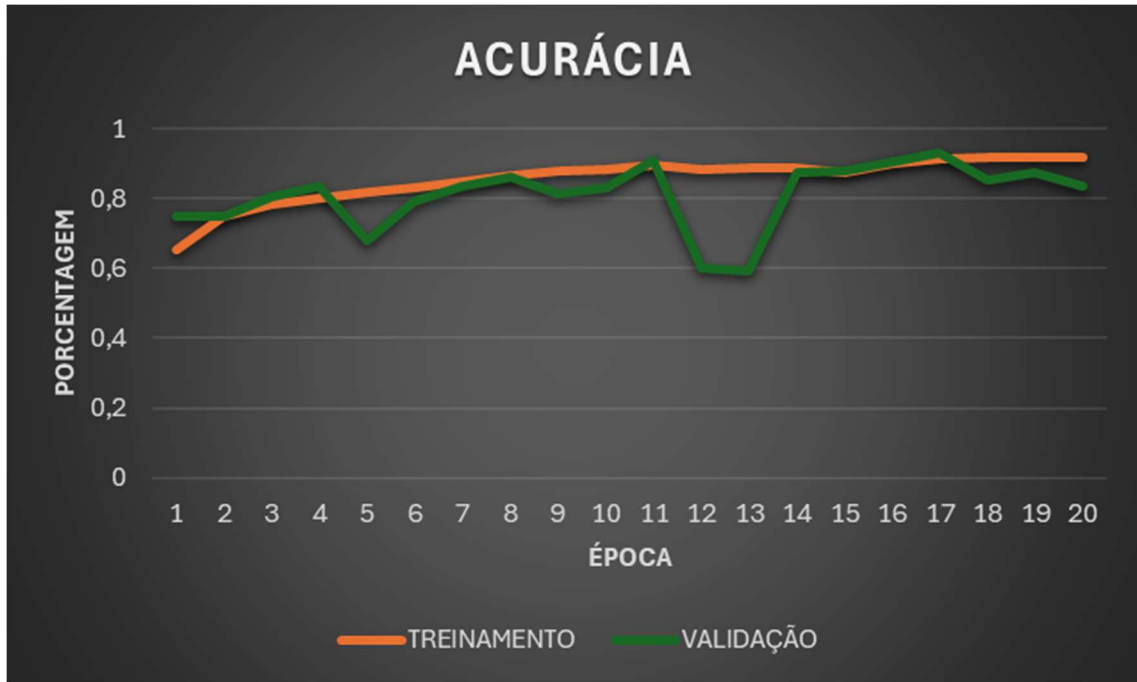


Figura 10 - Gráfico de Acurácia

A acurácia mede a proporção de previsões corretas feitas pelo modelo em relação ao total de previsões.

- **Train Accuracy:** A acurácia durante o treinamento apresentou uma melhoria contínua, alcançando 92% na 18ª época. Esse crescimento consistente sugere que o modelo está aprendendo a classificar os dados de treinamento com maior precisão ao longo do tempo.
- **Validation Accuracy:** Por outro lado, a acurácia de validação atingiu um pico de 83,83% na quarta época, antes de cair para 68,12% na quinta. As oscilações na acurácia de validação indicam que o modelo pode estar sensível às variações dos dados de validação, o que pode estar relacionado ao tamanho limitado do conjunto de validação ou à presença de alta variabilidade nos dados.

4.1.3. *Precision* (Precisão)

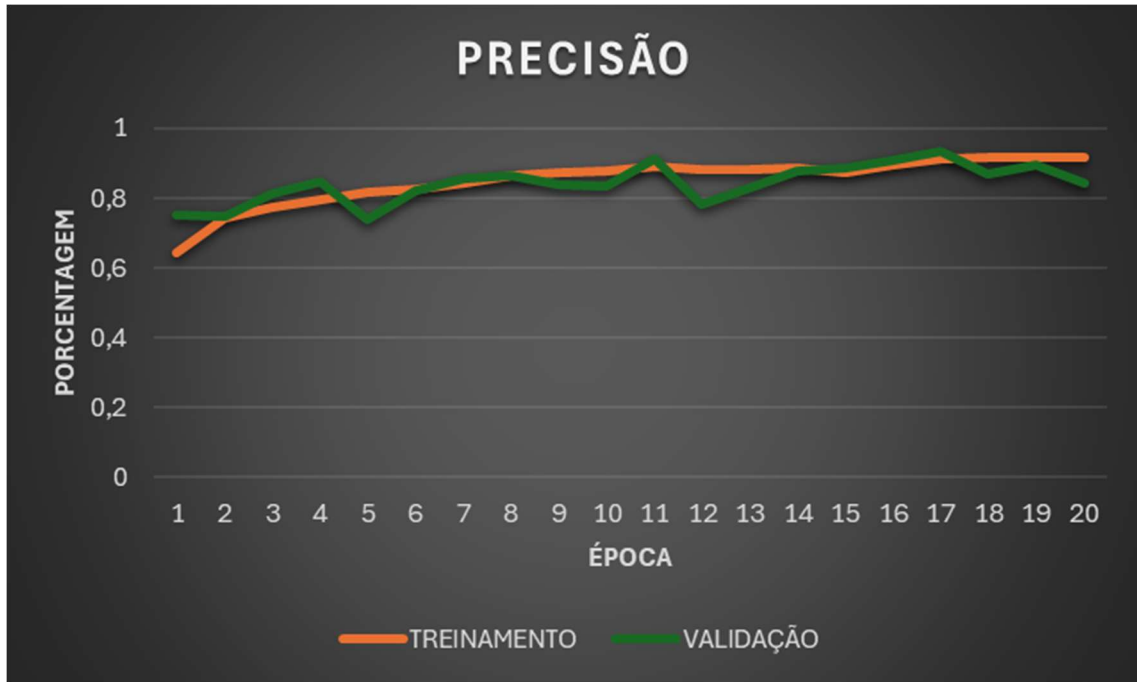


Figura 11 - Gráfico de Precisão

A precisão é uma métrica importante, especialmente em problemas onde o custo de falsos positivos é elevado. Ela mede a proporção de verdadeiros positivos entre todas as instâncias classificadas como positivas pelo modelo.

- **Train Precision:** A precisão durante o treinamento mostrou um crescimento estável, atingindo 91% na 17ª época. Isso demonstra que, ao longo do treinamento, o modelo foi capaz de prever corretamente as classes positivas com maior frequência.
- **Validation Precision:** A precisão de validação variou de maneira semelhante a acurácia, atingindo 84,58% na quarta época, mas caindo para 73,95% na quinta. Esta variação sugere que, embora o modelo seja geralmente preciso nas previsões, ele não mantém essa precisão de forma consistente em todas as épocas.

4.1.4. Hamming Loss



Figura 12 - Gráfico de Hamming Loss

A *Hamming Loss* é uma métrica que indica a taxa de erro do modelo, sendo que valores menores indicam melhor desempenho.

- **Train Hamming Loss:** A perda de *Hamming* durante o treinamento apresentou uma redução constante, de 0.3477 para 0.1794 ao longo das épocas, indicando uma melhora na capacidade do modelo em classificar corretamente as classes do conjunto de treinamento.
- **Validation Hamming Loss:** A perda de *Hamming* na validação, no entanto, mostrou flutuações mais significativas, atingindo 0.3188 na quinta época. Essa variação reforça a hipótese de que o modelo pode estar enfrentando dificuldades em manter uma classificação consistente nos dados de validação, possivelmente devido a um *overfitting* inicial.

4.1.5. RMSE (Root Mean Squared Error)

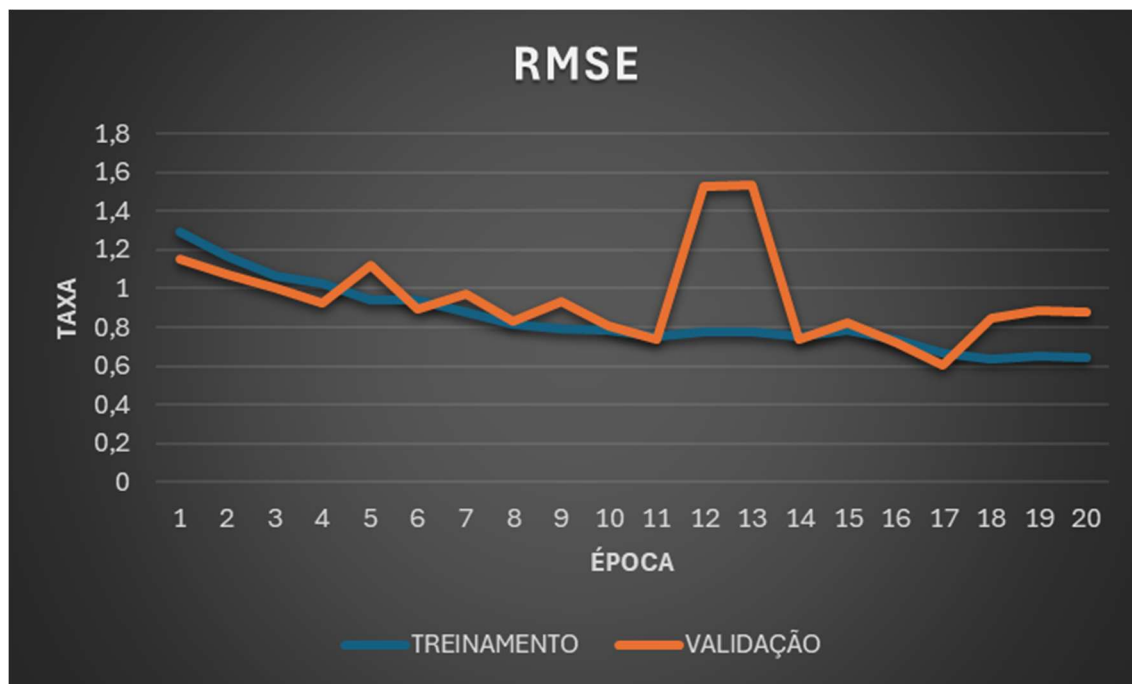


Figura 13 - Gráfico de RMSE

O RMSE mede a magnitude do erro das previsões, com valores menores indicando uma melhor precisão do modelo.

- **Train RMSE:** Durante o treinamento, o RMSE reduziu-se de 1.2937 para 0.9451, o que indica que o modelo foi capaz de melhorar a precisão de suas previsões ao longo do tempo.
- **Validation RMSE:** Para a validação, o RMSE inicialmente melhorou, mas apresentou um aumento na quinta época (1.1236), o que pode ser outro indicativo de *overfitting*, onde o modelo começa a se ajustar demasiadamente aos dados de treinamento, perdendo a capacidade de generalizar para novos dados.

4.2. Avaliação Geral do Modelo Proposto

Os resultados obtidos mostram que o modelo tem potencial, especialmente na fase de treinamento, onde as métricas de desempenho indicam um aprendizado efetivo. No entanto, as oscilações observadas nas métricas de validação sugerem que o modelo pode estar começando a super ajustar aos dados de treinamento, o que pode comprometer sua capacidade de generalização.

Essas flutuações são comuns em conjuntos de dados menores ou em situações onde há alta variabilidade nos dados de validação. Portanto, apesar dos resultados serem promissores, é importante considerar estratégias para mitigar os sinais de *overfitting* observados.

4.3. Propostas de Melhoria

Com base na análise dos resultados, algumas melhorias podem ser sugeridas para o aperfeiçoamento do modelo:

- **Regularização Adicional:** Implementar técnicas de regularização, como *dropout*, pode ajudar a reduzir a variabilidade nos resultados de validação, estabilizando o treinamento e minimizando o *overfitting*.
- **Aumento do *Dataset*:** Aumentar o tamanho do conjunto de dados, seja por coleta de mais dados ou por técnicas de data *augmentation*, pode ajudar a suavizar as flutuações observadas e melhorar a generalização do modelo.
- **Early Stopping:** Implementar um mecanismo de *early stopping* baseado em métricas como *Hamming Loss* ou RMSE pode ser eficaz para interromper o treinamento antes que o *overfitting* se torne predominante.

Com base nos resultados discutidos e nas melhorias propostas, torna-se evidente que o modelo desenvolvido apresenta um desempenho promissor, mas que ainda há espaço para refinamentos adicionais. No próximo capítulo, serão abordadas as conclusões finais do trabalho, bem como sugestões para futuros desenvolvimentos, com o intuito de aprimorar ainda mais a eficácia do modelo proposto e explorar suas aplicações práticas no campo da medicina.

5. CONCLUSÃO

Neste capítulo, apresentamos as conclusões finais derivadas do desenvolvimento e análise do modelo de Rede Neural Convolucional (CNN) para a classificação de tumores cerebrais. Além disso, discutimos as possíveis direções para trabalhos futuros, considerando tanto as melhorias propostas quanto as implicações positivas que essas tecnologias podem ter no campo da medicina.

O desenvolvimento deste trabalho demonstrou a viabilidade de utilizar redes neurais convolucionais para a classificação de tumores cerebrais, com um modelo que, apesar das

limitações encontradas, mostrou-se eficaz em diversas métricas de desempenho. A implementação da CNN personalizada resultou em uma curva de aprendizado positiva, com melhorias contínuas em métricas como acurácia, precisão e *Hamming Loss* ao longo do treinamento.

Entretanto, as oscilações observadas nas métricas de validação, especialmente nas últimas épocas de treinamento, indicam a necessidade de estratégias para combater o *overfitting*. As propostas de regularização adicional, aumento do conjunto de dados e a adoção de técnicas como *early stopping* são passos necessários para garantir que o modelo mantenha sua capacidade de generalização quando aplicado a novos dados.

As implicações deste estudo são significativas para o campo da medicina, particularmente na área de diagnóstico por imagem. A automatização do processo de classificação de tumores cerebrais não só tem o potencial de aumentar a precisão diagnóstica, mas também de reduzir o tempo necessário para alcançar um diagnóstico, o que é crucial para o tratamento precoce e eficaz de condições neurológicas. Além disso, a utilização de CNNs pode contribuir para a padronização dos diagnósticos, minimizando a variabilidade entre os profissionais de saúde e melhorando a qualidade do atendimento médico.

5.1. Trabalhos Futuros

A partir dos resultados obtidos e das limitações identificadas, algumas direções para trabalhos futuros podem ser sugeridas:

- **Aprimoramento do Modelo:** Implementar técnicas de regularização mais robustas, como dropout e batch normalization, para combater o overfitting observado nas fases finais de treinamento. Além disso, explorar diferentes arquiteturas de redes neurais que possam capturar de maneira mais eficiente as características relevantes das imagens médicas.
- **Expansão do *Dataset*:** Uma importante linha de investigação futura envolve a expansão do conjunto de dados utilizado. A coleta de um *dataset* mais amplo e diversificado, possivelmente incluindo imagens de diferentes fontes e variações de tumores, pode ajudar a melhorar a robustez do modelo e sua aplicabilidade em cenários reais.
- **Integração de Técnicas Avançadas:** A inclusão de técnicas avançadas, como redes neurais recorrentes (RNNs) para sequências temporais de imagens ou a aplicação de

redes generativas adversariais (GANs) para *data augmentation*, pode abrir novas possibilidades para a melhoria do modelo.

- **Aplicação Clínica e Validação:** Para consolidar a aplicação do modelo na prática clínica, seria necessário realizar estudos de validação em ambientes reais, envolvendo profissionais de saúde. A integração do modelo em sistemas de suporte à decisão clínica poderia ser testada em hospitais e clínicas, para avaliar seu impacto direto na rotina de diagnóstico e tratamento.
- **Exploração de Diagnósticos Multimodais:** Considerando a complexidade dos casos clínicos, uma abordagem multimodal que combine dados de imagem com informações clínicas, como histórico médico e dados genéticos, poderia ser explorada. Isso potencialmente aumentaria a precisão e a personalização dos diagnósticos.

5.2. Implicações para o Campo da Medicina

As redes neurais convolucionais oferecem uma ferramenta poderosa para o campo da medicina, especialmente na área de diagnóstico por imagem. A capacidade dessas redes de aprender e generalizar padrões complexos a partir de grandes volumes de dados abre novas perspectivas para a detecção precoce e precisa de doenças graves, como tumores cerebrais. A implementação prática desses modelos em sistemas de saúde pode não só melhorar a qualidade do diagnóstico, mas também aumentar a eficiência dos profissionais, permitindo que se concentrem em casos mais complexos e no planejamento de tratamentos.

Além disso, a adoção de tecnologias de inteligência artificial em processos diagnósticos pode levar a uma padronização mais rigorosa e a uma redução na variabilidade entre diferentes profissionais, contribuindo para uma maior equidade no atendimento ao paciente. Como resultado, os pacientes podem receber tratamentos mais rápidos e adequados, melhorando suas chances de recuperação e a qualidade de vida.

Em conclusão, este trabalho contribui para o avanço da inteligência artificial no campo da medicina, propondo soluções que, com os aprimoramentos sugeridos, têm o potencial de transformar o diagnóstico e o tratamento de doenças neurológicas, tornando-os mais precisos, eficientes e acessíveis.

REFERÊNCIAS

ARRIETA, A. B.; DÍAZ-RODRÍGUEZ, N.; SER, J. D.; BENNETOT, A.; TABIK, S.; BARBADO, A.; GARCÍA, S.; GIL-LÓPEZ, S.; MOLINA, D.; BENJAMINS, R.; et al. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, v. 58, p. 82–115, 2020. Disponível em: <https://arxiv.org/pdf/1910.10045>. Acesso em: 31 jul. 2024.

BISHOP, Christopher M. *Pattern recognition and machine learning*. New York: Springer, 2006.

BOKHARI, S. A.; ELNOOR, M.; AL MANSOUR, A.; MUSTAFA, K. Neuropsychiatric Manifestations of a Frontal Lobe Meningioma: A Case Report. *Cureus*, 2024. Disponível em: <https://www.cureus.com/articles/287096-neuropsychiatric-manifestations-of-a-frontal-lobe-meningioma-a-case-report.pdf>. Acesso em: 3 set. 2024.

BRAIN TUMOR EDUCATION. American Brain Tumor Association. Disponível em: https://www.abta.org/about-brain-tumors/brain-tumor-education/?gclid=CjwKCAjwko21BhAPEiwAwfaQCMxrfSNqJlrR_Lz1GwIanUiZRSv0nQLGjRPUqYgArMyXw7YSuEld4hoC_LUQAvD_BwE. Acesso em: 31 jul. 2024.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. Deep learning. Cambridge: MIT Press, 2016. Disponível em: <https://www.deeplearningbook.org/>. Acesso em: 13 ago. 2024

HUSSAIN, S. K.; RAMAY, S. A.; SHAHEER, H.; ABBAS, T. Automated Classification of Ophthalmic Disorders Using Color Fundus Images. *Kurdish Studies*, 2024. Disponível em: <https://kurdishstudies.net/menu-script/index.php/KS/article/download/3153/2123>. Acesso em: 3 set. 2024.

KURZWEIL, R. The age of intelligent machines. Cambridge, MA: MIT Press, 1990.

KURZWEIL, R. The singularity is near. New York: Viking, 2005.

LECUN, Yann; BENGIO, Yoshua. Convolutional networks for images, speech, and time series. In: *The handbook of brain theory and neural networks*. 1995. Disponível em: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=e26cc4a1c717653f323715d751c8dea7461aa105>. Acesso em: 31 jul. 2024.

LUSTOZA, L.; CAVALCANTI, W. Câncer de cérebro: dados apontam mais de 11 mil novos casos ao ano. Agência Saúde-DF, 6 maio 2024. Disponível em: <https://www.saude.df.gov.br/noticias/cancer-de-cerebro-dados-apontam-mais-de-11-mil-novos-casos-ao-ano/>. Acesso em: 31 jul. 2024.

NAYAK, D.R.; PADHY, N.; MALLICK, P.K.; ZYMBLER, M.; KUMAR, S. Brain Tumor Classification Using Dense Efficient-Net. *Axioms*, v. 11, n. 1, p. 34, 2022. Disponível em: <https://www.mdpi.com/2075-1680/11/1/34#metrics>. Acesso em: 31 jul. 2024.

NICKPARVAR, Masoud. Brain tumor MRI dataset. 2020. Disponível em: <https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset>. Acesso em: 3 set. 2024.

RUSSELL, S. J.; NORVIG, P. Inteligência artificial. Tradução de Regina Célia Simille. Rio de Janeiro: Elsevier, 2013.

SATHYA, R.; ABRAHAM, J. The proposed CNN model architecture. In: Comparative Study of Human and Machine Learning with Brain and Artificial Neural Network Models. 2019. Disponível em: https://www.researchgate.net/figure/The-proposed-CNN-model-architecture_fig2_336805909. Acesso em: 3 set. 2024.

SILVA, Rodrigo Emerson Valentim da. Um estudo comparativo entre redes neurais convolucionais para a classificação de imagens. 2018. 51 f. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) – Universidade Federal do Ceará, Campus de Quixadá, Quixadá, 2018. Disponível em: <https://repositorio.ufc.br/handle/riufc/39475>. Acesso em: 31 jul. 2024.

XIE, Y.; ZACCAGNA, F.; RUNDO, L.; TESTA, C.; AGATI, R.; LODI, R.; MANNERS, D.N.; TONON, C. Convolutional Neural Network Techniques for Brain Tumor Classification (from 2015 to 2022): Review, Challenges, and Future Perspectives. *Diagnostics*, v. 12, n. 8, p. 1850, 2022. Disponível em: <https://www.mdpi.com/2075-4418/12/8/1850#metrics>. Acesso em: 31 jul. 2024.

XU, Y.; JING, L.; YANG, W.; ZHANG, Y.; WANG, S.; BHARWANI, R. V. Deep Neural Networks for Brain Tumor Classification and Segmentation. ResearchGate, 2024. Disponível em: https://www.researchgate.net/profile/Kira-Sam/publication/382219837_Deep_Neural_Networks_for_Brain_Tumor_Classification_and_Segmentation/links/66923ff2af9e615a15e318c8/Deep-Neural-Networks-for-Brain-Tumor-Classification-and-Segmentation.pdf. Acesso em: 3 set. 2024.

YEO, Kee Kiat. Considerações gerais sobre tumores cerebrais e na medula espinhal em crianças. Disponível em: <https://www.msmanuals.com/pt-br/casa/problemas-de-sa%C3%BAdede-infantil/c%C3%A2nceres-na-inf%C3%A2ncia/considera%C3%A7%C3%B5es-gerais-sobre-tumores-cerebrais-e-na-medula-espinhal-em-crian%C3%A7as>. Acesso em: 31 jul. 2024.

