UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

EDUARDO MACEDO CAVALCANTI FREITAS

PODA ESTRUTURADA DE REDES NEURAIS PROFUNDAS: UM ESTUDO SOBRE A UTILIZAÇÃO DE MÉTODOS DE EXPLICABILIDADE COMO CRITÉRIO DE PODA

CAMPINA GRANDE - PB

2023

EDUARDO MACEDO CAVALCANTI FREITAS

PODA ESTRUTURADA DE REDES NEURAIS PROFUNDAS: UM ESTUDO SOBRE A UTILIZAÇÃO DE MÉTODOS DE EXPLICABILIDADE COMO CRITÉRIO DE PODA

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador : Herman Martins Gomes

CAMPINA GRANDE - PB

2023

EDUARDO MACEDO CAVALCANTI FREITAS

PODA ESTRUTURADA DE REDES NEURAIS PROFUNDAS: UM ESTUDO SOBRE A UTILIZAÇÃO DE MÉTODOS DE EXPLICABILIDADE COMO CRITÉRIO DE PODA

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA:

Herman Martins Gomes

Orientador – UASC/CEEI/UFCG

Patrícia Duarte de Lima Machado

Examinador – UASC/CEEI/UFCG

Francisco Vilar Brasileiro

Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 28 de JUNHO de 2023.

CAMPINA GRANDE - PB

RESUMO

O avanço dos modelos de Deep Learning tem proporcionado resultados excepcionais em tarefas de visão computacional e processamento de linguagem natural. No entanto, o aumento do tamanho e complexidade desses modelos traz desafios significativos em termos de infraestrutura e custos operacionais. Nesse contexto, a técnica de poda em redes neurais profundas surge como uma solução para reduzir o tamanho dos modelos, mantendo níveis similares de acurácia. Este estudo investiga o impacto da utilização de métricas de explicabilidade (Conductance e Layer-wise Relevance Propagation) como critério de poda, comparando-as com a poda por magnitude de pesos e a poda aleatória. Diferentes percentuais de poda são avaliados, considerando tanto a poda de oneshot quanto a poda iterativa. Os resultados mostram uma correlação positiva entre o uso de métricas de explicabilidade e a melhoria na qualidade dos modelos podados, incluindo maior acurácia, menor variância e a capacidade de realizar podas mais agressivas sem perda significativa de acurácia. Esses métodos promissores têm o potencial de melhorar a operacionalização e reduzir os custos associados aos modelos de Deep Learning em larga escala.

STRUCTURED PRUNING IN DEEP NEURAL NETWORKS: A STUDY ON THE USE OF EXPLAINABILITY METHODS AS PRUNING CRITERIA

ABSTRACT

The advancement of Deep Learning models has provided exceptional results in computer vision and natural language processing tasks. However, the increase in size and complexity of these models brings significant challenges in terms of infrastructure and operational costs. In this context, the technique of structured pruning in deep neural networks emerges as a solution to reduce model size while maintaining similar levels of accuracy. This study investigates the impact of using explainability metrics (Conductance and Layer-wise Relevance Propagation) as pruning criteria, comparing them with magnitude-based pruning and random pruning. Different pruning percentages are evaluated, considering both one-shot pruning and iterative pruning. The results show a positive correlation between the use of explainability metrics and improvement in the quality of pruned models, including higher accuracy, lower variance, and the ability to perform more aggressive pruning without significant loss of accuracy. These promising methods have the potential to enhance operationalization and reduce costs associated with large-scale Deep Learning models.

Poda Estruturada em Redes Neurais Profundas: Um Estudo sobre a Utilização de Métodos de Explicabilidade como Critério de Poda

Eduardo Macedo Cavalcanti Freitas
Universidade Federal de Campina Grande
Campina Grande, Paraíba - Brasil
eduardo.freitas@ccc.ufcg.edu.br

Herman Martins Gomes (Orientador)
Universidade Federal de Campina Grande
Campina Grande, Paraíba - Brasil
hmg@computacao.ufcg.edu.br

RESUMO

O avanço dos modelos de Deep Learning tem proporcionado resultados excepcionais em tarefas de visão computacional e processamento de linguagem natural. No entanto, o aumento do tamanho e complexidade desses modelos traz desafios significativos em termos de infraestrutura e custos operacionais. Nesse contexto, a técnica de poda em redes neurais profundas surge como uma solução para reduzir o tamanho dos modelos, mantendo níveis similares de acurácia. Este estudo investiga o impacto da utilização de métricas de explicabilidade (*Conductance* e *Layer-wise Relevance Propagation*) como critério de poda, comparando-as com a poda por magnitude de pesos e a poda aleatória. Diferentes percentuais de poda são avaliados, considerando tanto a poda de *oneshot* quanto a poda iterativa. Os resultados mostram uma correlação positiva entre o uso de métricas de explicabilidade e a melhoria na qualidade dos modelos podados, incluindo maior acurácia, menor variância e a capacidade de realizar podas mais agressivas sem perda significativa de acurácia. Esses métodos promissores têm o potencial de melhorar a operacionalização e reduzir os custos associados aos modelos de Deep Learning em larga escala.

Palavras-Chave

Poda Estruturada, Deep Learning, Explicabilidade.

1. INTRODUÇÃO

Modelos de *Deep Learning* têm atingido resultados estado-da-arte em uma variedade de tarefas relacionadas à visão computacional e processamento de linguagem natural. Esse avanço influenciou o surgimento de ferramentas como assistentes virtuais inteligentes, tradutores automáticos avançados, sistemas de recomendação personalizados, dentre outras. Tais ferramentas têm se disseminado cada vez mais, transformando a maneira como interagimos com a tecnologia e impactando significativamente a vida das pessoas.

Tal sucesso pode ser creditado, dentre outros fatores, à descoberta de arquiteturas mais robustas [8, 21], à disponibilidade de conjuntos de dados massivos e ao aumento da quantidade de parâmetros dos modelos - uma vez que esses fatores conferem aos modelos uma maior capacidade de aprendizado e generalização.

No entanto, o crescimento vertiginoso no tamanho e na complexidade dos modelos de *Deep Learning* acarreta desafios significativos em termos de infraestrutura e custos operacionais envolvidos no treinamento e no processo de inferência desses modelos. Para lidar com modelos cada vez maiores e mais complexos, é necessária uma infraestrutura computacional poderosa, com diversas unidades de processamento gráfico (GPUs) e/ou unidades de processamento tensorial (TPUs), para garantir o processamento eficiente dos dados e a aceleração do treinamento. O processo de treinamento também pode se estender por semanas ou até meses, ampliando os custos operacionais associados e incumbindo em maiores gastos energéticos e emissões de carbono [16, 19].

Os desafios mencionados anteriormente suscitam questões relevantes sobre a viabilidade econômica e sustentabilidade do treinamento e inferência de modelos de *Deep Learning* em larga escala. No entanto, estão surgindo soluções para mitigar esses custos, como a técnica de poda em redes neurais profundas. A poda visa remover conexões redundantes ou menos relevantes nos modelos, resultando na redução do tamanho dos mesmos, enquanto mantém níveis de acurácia comparáveis ou até superiores [5]. Essa redução no tamanho dos modelos traz benefícios significativos, como a diminuição dos custos operacionais, de infraestrutura e do impacto ambiental associado ao processo de treinamento e inferência de modelos.

No entanto, para realizar a poda dos modelos, é necessário estabelecer critérios que determinem quais estruturas da rede (neurônios e/ou filtros) serão preservadas e quais serão removidas. Inicialmente, a magnitude dos pesos era a métrica utilizada como critério de seleção. Contudo, estudos recentes têm proposto a utilização de métodos de explicabilidade de redes neurais como critério de filtragem [1, 3].

A adoção de métricas de explicabilidade na poda de modelos de redes neurais oferece diversas vantagens significativas. Essas métricas permitem uma compreensão mais profunda do funcionamento interno dos modelos, identificando as conexões e unidades de processamento mais relevantes para as decisões tomadas pelo modelo. Além de fundamentar e tornar o processo de poda mais interpretável, a incorporação de métricas de explicabilidade pode melhorar a seleção das estruturas a serem preservadas ou removidas, resultando em uma poda mais seletiva e eficiente. Isso não apenas reduz o tamanho dos modelos, mas também ajuda a melhorar a interpretabilidade dos mesmos, o que é particularmente valioso em aplicações críticas, como medicina, direito e finanças.

Nesse contexto, o presente estudo tem como objetivo investigar o impacto da utilização de métricas de explicabilidade como critério

para a poda estruturada em redes neurais profundas. Para alcançar esse objetivo, as métricas de explicabilidade são comparadas com a poda por magnitude dos pesos e com a poda aleatória. São definidos diversos percentuais de poda a fim de avaliar a eficiência de cada critério estabelecido. Em todos os casos, aplicou-se uma poda local, ou seja, o percentual de poda é aplicado separadamente em cada camada da rede neural. Além disso, os impactos de diferentes abordagens de poda, como a poda de *oneshot* e a poda iterativa, também são avaliados.

Conclusivamente, os resultados obtidos nesta pesquisa indicam uma forte correlação entre a aplicação de métricas de explicabilidade como critério de poda e a melhoria da qualidade dos modelos podados. Essa melhoria se reflete em termos de maior acurácia, menor variância e a possibilidade de utilizar percentuais mais elevados de poda com uma penalização mínima na acurácia. Portanto, a utilização desses métodos apresenta-se promissora para aprimorar a operacionalização e reduzir os custos associados a modelos de Deep Learning de grande escala.

2. FUNDAMENTAÇÃO TEÓRICA

Esta seção fornece detalhes dos fundamentos teóricos necessários para um completo entendimento da pesquisa. Nela são abordados conceitos relacionados à poda de redes neurais, bem como a métricas de explicabilidade.

2.1 Tipos de poda

Existem dois tipos de poda: não-estruturada e estruturada. Eles variam conforme as estruturas que serão removidas e ambas apresentam vantagens e desvantagens.

2.1.1 Não-estruturada

Na poda não-estruturada, também conhecida como poda de pesos individuais, as conexões individuais da rede neural são avaliadas com base em um critério estabelecido, frequentemente a magnitude dos pesos. Conexões com valores de peso abaixo de um limiar definido são removidas, enquanto as demais são mantidas. Essa abordagem pode resultar em uma rede neural irregular e desconectada, tornando a implementação em hardware especializado e a aplicação de técnicas adicionais de compressão mais desafiadoras.

2.1.2 Estruturada

Por outro lado, na poda estruturada, também chamada de poda por camada ou poda por estrutura, unidades inteiras de processamento, como neurônios ou filtros, são removidas. Essa abordagem preserva a estrutura regular da rede neural, resultando em um padrão de esparsidade mais organizado. A poda estruturada facilita a implementação eficiente em hardware especializado, tirando proveito da regularidade estrutural para obter benefícios de otimização. Além disso, a remoção de unidades completas pode simplificar a interpretação do modelo, permitindo identificar unidades irrelevantes.

2.2 Localidade da poda

Quanto à localidade, duas abordagens comuns são a poda local e a poda global. Elas diferem na forma como as conexões são avaliadas e removidas.

2.2.1 Local

Na poda local, a remoção das conexões ocorre individualmente em cada camada da rede neural. Nessa abordagem, as conexões menos importantes ou redundantes são identificadas e eliminadas dentro de cada camada, de forma proporcional.

2.2.2 Global

Na poda global, os elementos são eliminados independentemente de sua localização na rede neural. Isso significa que, para atingir uma taxa de poda específica, diferentes frações dessa taxa são aplicadas em cada camada com base em uma classificação global dos elementos da rede neural. É importante salientar que, a depender da agressividade da taxa de poda e do critério de filtragem de conexões, existe a possibilidade da poda global remover todas as conexões de uma camada, inviabilizando completamente a rede.

2.3 Temporalidade da poda

A temporalidade da poda é um aspecto fundamental na aplicação dessa técnica de compressão de redes neurais. Ela se refere à maneira como a poda é realizada ao longo do processo de treinamento da rede neural. Existem três abordagens principais de temporalidade de poda: poda *oneshot*, poda iterativa e poda gradual.

2.3.1 Oneshot

Na poda *oneshot*, a poda é aplicada à rede neural treinada em um único momento, geralmente ao final do processo de treinamento. Nesse método, a taxa de poda desejada é estabelecida e as conexões são removidas de forma a atingir essa taxa. Após a poda, pode ser necessária uma etapa adicional de retreinamento da rede podada para recuperar o desempenho perdido devido à remoção das conexões.

2.3.2 Iterativa

A poda iterativa, por sua vez, envolve a aplicação da poda em pequenas frações ao final de cada ciclo de treinamento. Esse procedimento visa atingir a taxa de poda desejada ao longo de múltiplas iterações do treinamento e do processo de poda. Após cada iteração, ocorre o retreinamento da rede neural, com o intuito de mitigar a perda de desempenho decorrente da poda. Esse ciclo de poda e retreinamento é repetido até que a taxa de poda almejada seja alcançada.

2.3.3 Gradual

Na abordagem da poda gradual, ocorre a remoção contínua de conexões menos relevantes ou redundantes da rede neural ao longo de todo o processo de treinamento. À medida que o treinamento avança, essas conexões são identificadas e eliminadas, resultando em uma rede neural já podada ao final do ciclo de treinamento. Diferentemente de outras abordagens, como a poda oneshot e a poda iterativa, a poda gradual não requer uma etapa adicional de retreinamento, uma vez que o processo de poda ocorre de forma simultânea ao treinamento.

2.4 Rebobinamento de pesos

O rebobinamento de pesos, também conhecido na língua inglesa como *weight rewinding* ou *weight reset*, é uma técnica utilizada no contexto de poda de redes neurais para mitigar a perda de desempenho decorrente da remoção de conexões durante o processo de poda. Quando determinadas conexões são podadas com base em um critério predefinido, é possível que o desempenho da rede seja comprometido devido à perda de representações relevantes à tarefa.

Essa abordagem envolve o armazenamento dos valores dos pesos antes da poda e, em seguida, a atribuição desses valores aos pesos correspondentes após a poda. A restauração temporária dos pesos tem como objetivo preservar as informações originalmente presentes nas conexões podadas, com o intuito de minimizar o impacto negativo na performance da rede neural [5, 14].

2.5 Métodos de explicabilidade

A eficácia dos modelos de *Deep Learning* na resolução de tarefas muitas vezes vem acompanhada de uma falta de transparência, tornando-os caixas pretas, ou seja, com processo de tomada de decisão de difícil entendimento. Essa opacidade é uma preocupação significativa, especialmente em áreas onde a auditabilidade e a explicabilidade são essenciais. Para mitigar esse problema, esforços têm sido empreendidos no campo da Inteligência Artificial Explicável (XAI, do inglês Explainable Artificial Intelligence) [4, 6] para desenvolver métodos e técnicas que tornem o processo de tomada de decisão das redes neurais mais transparente, permitindo compreender as razões por trás de suas predições e fornecendo *insights* interpretáveis para os usuários finais e especialistas. Essa abordagem visa fornecer uma compreensão mais aprofundada das características, padrões e relações que influenciam as decisões do modelo, aumentando a confiança e a aceitação das redes neurais em aplicações críticas.

Métodos de explicabilidade fornecem esclarecimentos sobre o processo de tomada de decisão das redes neurais, permitindo compreender quais características ou partes da entrada de dados são mais relevantes para as predições feitas pelo modelo. Tais métodos atribuem um valor de importância para cada entrada, unidade ou camada do modelo, com base em sua contribuição para a predição final. No contexto de redes convolucionais (CNN) métodos de explicabilidade resultam em mapas de saliência, que destacam as regiões mais relevantes da entrada (como observado na Figura 1).

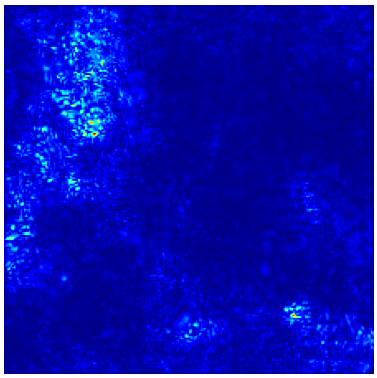

Figura 1: Um mapa de saliência onde os pixels da imagem estão coloridos de acordo com a sua contribuição para a classificação - regiões mais claras representam maior contribuição. [9]

Na presente pesquisa dois desses métodos foram empregados a fim de obter a importância de cada estrutura e utilizá-la como critério de poda: *Conductance* e *Layer-wise relevance propagation*, conforme discutido a seguir.

2.5.1 Conductance

A *conductance* é uma métrica de importância baseada na técnica de *Integrated Gradients* (IG) [20]. A técnica IG integra os gradientes da saída da predição em relação à entrada, considerando diferentes variantes da entrada, resultando em valores de importância para cada pixel de entrada.

A métrica de *conductance* quantifica a importância de uma unidade escondida (*hidden unit*) da rede neural como o fluxo de atribuição dos *Integrated Gradients* através dessa unidade, decompondo o cálculo dos IGs usando a regra da cadeia [3]. Essa métrica é especialmente útil para a poda, pois permite calcular a importância de estruturas específicas da rede, como neurônios e filtros em uma arquitetura de rede convolucional (CNN), em vez de avaliar a importância de pixels individuais.

De maneira análoga ao IG, é necessário fornecer entradas para a rede de modo a calcular a importância de cada *hidden unit*. A importância final consiste da média das importâncias para cada entrada distinta

2.5.2 Layer-wise relevance propagation (LRP)

Layer-wise relevance propagation é uma técnica utilizada em deep learning para compreender as contribuições de neurônios individuais e características de entrada para a saída da rede [1]. Esta técnica permite rastrear a relevância ou importância de cada neurônio e pixel na entrada da rede até a predição final. A LRP opera propagando valores de relevância da camada de saída de volta para a camada de entrada, fornecendo insights sobre os fatores que influenciaram o processo de tomada de decisão da rede.

A ideia central por trás da LRP é distribuir a relevância da saída entre os neurônios e pixels nas camadas anteriores com base em suas contribuições. Essa distribuição é realizada aplicando regras específicas durante o processo de retropropagação. Uma regra comum da LRP é a LRP-0, que calcula a relevância de cada neurônio em uma camada considerando as contribuições dos neurônios na camada subsequente. Esse processo envolve a multiplicação da relevância dos neurônios subsequentes pelos pesos de conexão e a divisão pelo somatório dos pesos de todas as conexões naquela camada.

A LRP mantém uma propriedade de "conservação", o que significa que os valores de relevância são preservados ao longo do processo de retropropagação. A relevância atribuída a um neurônio ou pixel na camada de entrada é igual à soma dos valores de relevância propagados pela camada de saída. Essa propriedade garante que a relevância total seja conservada e fornece uma maneira de atribuir relevância a cada componente da entrada.

3. METODOLOGIA

O presente trabalho consiste de um estudo qualitativo experimental. As seções seguintes detalham os procedimentos executados para a realização dos experimentos, bem como para a análise dos resultados. A seguir são apresentados o conjunto de dados, as etapas de pré-processamento, a arquitetura de rede, a orquestração de experimentos de poda e as métricas de avaliação.

3.1 Conjunto de dados

O conjunto de dados utilizado para treinar e validar os modelos foi o CIFAR-10 [10]. Esse conjunto é amplamente utilizado na área de redes neurais como referência para avaliar algoritmos de

aprendizado de máquina e visão computacional. É composto por 60.000 imagens coloridas de baixa resolução, divididas em 10 classes distintas, cada uma contendo 6.000 imagens. As classes incluem objetos como aviões, automóveis, pássaros, gatos, veados, cães, sapos, cavalos, navios e caminhões (Figura 2).

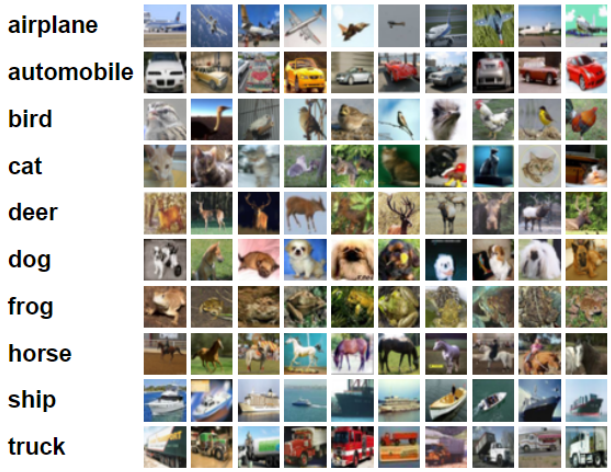

Figura 2: Ilustração das classes e imagens do conjunto de dados CIFAR-10

Uma das principais características do CIFAR-10 é a sua representatividade da diversidade visual do mundo real, tornando-o um desafio interessante para o treinamento de redes neurais. As imagens possuem uma resolução de 32x32 pixels, o que requer a extração de características significativas a partir de informações limitadas. Além disso, o conjunto de dados apresenta variações de iluminação, posição e escala, o que adiciona complexidade ao processo de treinamento.

O CIFAR-10 é frequentemente dividido em dois conjuntos: treino e teste. O conjunto de treino contém 50.000 imagens, que são utilizadas para treinar os modelos de rede neural. Já o conjunto de teste é composto por 10.000 imagens e é utilizado para avaliar a capacidade de generalização do modelo, ou seja, sua habilidade de classificar corretamente imagens que não foram vistas durante o treinamento. Essa divisão é importante para garantir uma avaliação imparcial do desempenho do modelo em dados não vistos anteriormente, ajudando a evitar problemas de overfitting.

As imagens do conjunto de dados foram pré-processadas por meio da normalização dos canais de cor. Esse processo envolveu o cálculo da média e do desvio padrão de cada canal, seguido pela subtração da média e divisão pelo desvio padrão correspondente para cada valor de pixel. Essa normalização desempenha um papel fundamental no treinamento da rede neural, pois impede que características com escalas maiores influenciem de forma desproporcional o aprendizado em relação a características com escalas menores. Além disso, a normalização contribui para a estabilização do gradiente durante a propagação do erro, resultando em uma convergência mais rápida e eficiente do modelo.

3.2 Arquitetura da rede

A arquitetura escolhida para a realização dos experimentos foi a MobileNetV2 [15]. Essa arquitetura apresenta resultados competitivos, mesmo com uma quantidade comparativamente menor de parâmetros. Isso se dá devido a particularidades da rede, pensadas para extrair o máximo de eficácia com o mínimo de recursos computacionais.

Uma dessas particularidades é o uso de um bloco chamado *Linear Bottleneck*, que permite a aprendizagem de uma representação compacta e eficiente. Esse bloco consiste em uma sequência de camadas, incluindo uma convolução 1x1, uma convolução separável 3x3 e outra convolução 1x1. A convolução 1x1 inicial tem o objetivo de reduzir a dimensão dos canais, enquanto a convolução separável 3x3 captura as informações relevantes. Por fim, a última convolução 1x1 aumenta a dimensionalidade novamente.

A MobileNetV2 também emprega uma técnica conhecida como *depthwise separable convolution* (conforme ilustra a Figura 3). Nessa técnica, a convolução espacial e a convolução em profundidade são realizadas separadamente. Primeiro, uma convolução espacial é aplicada para capturar informações espaciais, seguida por uma convolução em profundidade que lida com a dimensionalidade dos canais. Essa abordagem reduz significativamente a quantidade de operações necessárias, tornando a rede mais eficiente em termos computacionais.

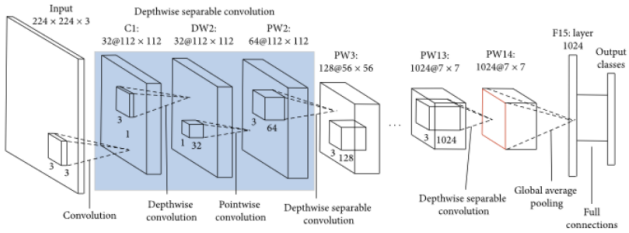

Figura 3:Arquitetura MobileNetV2

Por fim, é importante destacar que a MobileNetV2 não possui viés (*bias)* em suas camadas convolucionais. Essa escolha foi feita para reduzir o número total de parâmetros do modelo e melhorar sua eficiência computacional. Para compensar a ausência de *bias*, a MobileNetV2 utiliza a técnica de normalização em lote (batch normalization) após cada camada convolucional, garantindo a estabilidade do aprendizado.

3.3 Treinamento de um modelo de referência

A fim de estabelecer um ponto de referência para a avaliação dos diferentes métodos de poda e a viabilidade da estratégia de rebobinamento de pesos, um modelo de referência (*baseline*) foi treinado utilizando o conjunto de dados e a arquitetura mencionados anteriormente. Durante o processo de treinamento, foi adotado o algoritmo de otimização conhecido como *Stochastic Gradient Descent* (SGD). Para essa configuração específica, foi empregada uma taxa de aprendizagem fixa de 0.01, um valor de momentum de 0.9, uma taxa de decaimento de pesos de 0.0001 e um tamanho de *batch* de 128, parâmetros estabelecidos de acordo com a referência [12]. A duração do treinamento foi de 50 épocas.

3.4 Experimentos de poda

Com o objetivo de comparar e avaliar a efetividade dos métodos de poda, bem como investigar suas limitações, foram conduzidos

experimentos abrangentes que envolveram diversas configurações de poda.

3.4.1 Configuração dos experimentos

Os experimentos foram projetados considerando a aplicação de poda estruturada e local como abordagens preferenciais devido à ampla aceitação de que essas modalidades resultam em desempenho aprimorado em termos de acurácia de classificação [7, 12, 13, 22].

Além disso, foram realizados experimentos com diferentes abordagens temporais de poda, incluindo a poda *oneshot* e a poda iterativa. Foram selecionados cinco níveis distintos de porcentagem de poda: 0.5, 0.6, 0.7, 0.8 e 0.9. Quanto aos critérios de filtragem de peso, foram empregados os métodos de filtragem randômica, filtragem por magnitude, filtragem por *conductance* e filtragem por LRP. Para mitigar o impacto da aleatoriedade nos resultados, cada combinação de parâmetros foi executada três vezes. Em todos os experimentos, a técnica de rebobinamento de pesos foi aplicada após o processo de poda, a fim de restabelecer a funcionalidade do modelo e mitigar possíveis perdas de desempenho.

A estrutura selecionada para ser submetida ao processo de poda foram os filtros das camadas de convolução 2D. A poda foi realizada de acordo com a seguinte abordagem: inicialmente, aplicou-se uma regularização L1 nos valores de importância atribuídos a cada filtro. Em seguida, calculou-se a média dos valores de importância para cada filtro, e os filtros com menor importância foram removidos. A quantidade de filtros removidos variou de acordo com o percentual de poda adotado.

Esses experimentos foram cuidadosamente projetados para fornecer uma análise abrangente da influência dos métodos de poda em termos de desempenho do modelo, explorando múltiplos cenários através das combinações dos parâmetros de poda. Tais investigações são fundamentais para compreender de forma mais aprofundada as características e potencialidades dessas técnicas de compressão de modelos de redes neurais.

3.4.2 Poda oneshot

Nos experimentos baseados em poda *oneshot*, cada combinação de configuração mencionada anteriormente foi submetida a um único estágio de poda. As peculiaridades de cada experimento variaram de acordo com o método de atribuição de importância selecionado, dividindo-se em poda *oneshot* convencional (que engloba a poda aleatória e por magnitude) e poda *oneshot* explicativa (que engloba a poda por *conductance* e LRP).

Na poda oneshot convencional, a seleção dos filtros a serem podados foi realizada com base em dois critérios amplamente utilizados: aleatoriedade e magnitude. Os filtros com menor importância, determinados por esses critérios, foram removidos. Tal processo é ilustrado no Algoritmo 1.

Algorithm 1 Poda Oneshot Aleatória ou Baseada na Magnitude dos Pesos

função PodaOneshotConvencional(m, t, p)
  entradas: Modelo treinado m; Taxa de poda t; Método de poda p
  saída: Modelo podado m'

  para cada camada convolucional $c_i \in m$ **faça**
    para cada filtro $f_i \in c_i$ **faça**
      calcular a norma L1 dos pesos de f_i;
    fim para cada
  fim para cada

  se $p == magnitude$ **então**
    ordenar normas por camada;
    podar os filtros com menores normas até alcançar a taxa t por camada;
  senão se $p == aleatorio$ **então**
    podar filtros aleatoriamente até alcançar a taxa t por camada;
  fim se
  retorna m'

Algoritmo 1: Descrição da poda *oneshot* utilizando métodos convencionais

Por outro lado, na poda oneshot explicativa, foram empregados critérios mais sofisticados, como *conductance* e LRP, que permitem uma avaliação mais detalhada da importância dos filtros. Essa modalidade conta com uma etapa adicional para o cálculo dos valores de importância de cada filtro. Os filtros foram selecionados com base nesses critérios e subsequentemente podados. O Algoritmo 2 descreve esse processo.

Algorithm 2 Poda Oneshot Baseada em Métodos de Explicabilidade

função PodaOneshotExplicativa(m, D, t, p)
  entradas: Modelo treinado m; Conjunto de dados D; Taxa de poda t; Método de poda p
  saída: Modelo podado m'

  para cada camada convolucional $c_i \in m$ **faça**
    para cada entrada $e_i \in D$ **faça**
      calcular os valores de importância através do método p;
    fim para cada

    para cada filtro $f_i \in c_i$ **faça**
      calcular a norma L1 dos valores de importância do método p;
    fim para cada
  fim para cada

  ordenar normas por camada;
  podar os filtros com menores normas até alcançar a taxa t por camada;
  retorna m'

Algoritmo 2: Descrição da poda *oneshot* utilizando métodos de explicabilidade

3.4.3 Poda iterativa

Os experimentos de poda iterativa seguiram uma abordagem semelhante à poda *oneshot*. Eles também foram divididos com base na metodologia de atribuição de importância aos filtros a serem podados, incluindo a poda iterativa convencional e a poda iterativa explicativa. No entanto, a distinção da poda iterativa explicativa reside nas iterações sucessivas de poda.

Nesse contexto, dado um percentual total de poda desejado, foi definido um percentual de poda por iteração com base no número de iterações selecionadas. Ao final do processo, a combinação das podas sucessivas resultou no percentual total de poda predefinido. Em todos os experimentos realizados, foram conduzidas três iterações de poda. Os algoritmos que representam a poda iterativa convencional e explicativa estão representados nos Algoritmos 3 e 4, respectivamente.

Algorithm 3 Poda Iterativa Aleatória ou Baseada na Magnitude dos Pesos

função PODAITERATIVACONVENCIONAL(m, t, p, i)
entradas: Modelo treinado m; Taxa de poda t; Método de poda p; Número de iterações i
saída: Modelo podado m'

$t' \leftarrow$ CalculaTaxaPodaPorIteracao(t, i) ▷ taxa de poda por iteração
para cada iteração i **faça**
  para cada camada convolucional $c_i \in m$ **faça**
    para cada filtro $f_i \in c_i$ **faça**
      calcular a norma L1 dos pesos de f_i
    fim para cada
  fim para cada

  se $p == magnitude$ **então**
    ordenar normas por camada;
    podar os filtros com menores normas até alcançar a taxa t' por camada;
  senão se $p == aleatorio$ **então**
    podar filtros aleatoriamente até alcançar a taxa t' por camada;
  fim se
fim para cada

retorna m'

Algoritmo 3: Descrição da poda iterativa utilizando métodos convencionais

Algorithm 4 Poda Iterativa Baseada em Métodos de Explicabilidade

função PODAITERATIVAEXPLICATIVA(m, D, t, p, i)
entradas: Modelo treinado m; Conjunto de dados D; Taxa de poda t; Método de poda p; Número de iterações i
saída: Modelo podado m'

$t' \leftarrow$ CalculaTaxaPodaPorIteracao(t, i) ▷ taxa de poda por iteração
para cada iteração i **faça**
  para cada camada convolucional $c_i \in m$ **faça**
    para cada entrada $e_i \in D$ **faça**
      calcular os valores de importância através do método p;
    fim para cada

    para cada filtro $f_i \in c_i$ **faça**
      calcular a norma L1 dos valores de importância do método p;
    fim para cada
  fim para cada

  ordenar normas por camada;
  podar os filtros com menores normas até alcançar a taxa t' por camada;
fim para cada

retorna m'

Algoritmo 4: Descrição da poda iterativa utilizando métodos de explicabilidade

Além disso é importante pontuar que a etapa de rebobinamento de pesos e retreinamento fora aplicada após cada iteração de poda, com o propósito de manter a capacidade de generalização do modelo, preservando a representação dos padrões aprendidos anteriormente e permitindo que o modelo se adapte às mudanças introduzidas pelas iterações de poda.

3.5 Métrica de avaliação

No intuito de avaliar o desempenho dos modelos submetidos à poda, a métrica selecionada para análise foi a acurácia. A acurácia foi considerada uma métrica apropriada para esse propósito devido às características do conjunto de dados utilizado, no qual todas as classes estão balanceadas, ou seja, apresentam um número igual de exemplos. Nesse contexto, a acurácia não sofre de viés, como ocorreria em situações com desbalanceamento de classes.

Além disso, a variação de desempenho de cada modelo também foi considerada para determinar a consistência dos resultados. Essa análise levou em conta a variância dos modelos, ou seja, a dispersão dos valores obtidos ao realizar múltiplas execuções dos experimentos. Com base nessa avaliação da variância, foi possível identificar quais modelos apresentaram maior consistência e estabilidade em seus resultados.

Dessa forma, a escolha da métrica de acurácia e a consideração da variação dos modelos permitiram uma avaliação abrangente do desempenho dos modelos podados, levando em consideração tanto a acurácia geral quanto a consistência dos resultados. Essa abordagem contribuiu para a obtenção de *insights* mais robustos e confiáveis sobre a efetividade dos métodos de poda utilizados.

4. RESULTADOS

4.1 Baseline

Os resultados obtidos por meio da metodologia descrita podem ser divididos em três categorias: *baseline*, poda oneshot e poda iterativa. O modelo *baseline*, conforme as configurações de treinamento mencionadas anteriormente, alcançou uma acurácia de 0.7539. Esse resultado é utilizado como referência para avaliar a eficácia da poda.

4.2 Poda *oneshot*

A Figura 4 apresenta, para cada taxa de poda experimentada, as variações de acurácia dos métodos avaliados. É possível notar que o método de poda baseado em *conductance* apresentou consistentemente um desempenho superior em comparação aos demais métodos. Mesmo em taxas de poda agressivas, esse método alcançou resultados satisfatórios e demonstrou uma variância significativamente menor em relação aos demais. A poda por magnitude superou a poda por lrp até o percentual de 0.8, entretanto, acima desse valor, a poda por magnitude resultou em uma queda abrupta na acurácia do modelo. Por outro lado, a poda por lrp mostrou-se capaz de superar a poda por magnitude no percentual de poda de 90%, embora tenha apresentado uma variância extremamente alta.

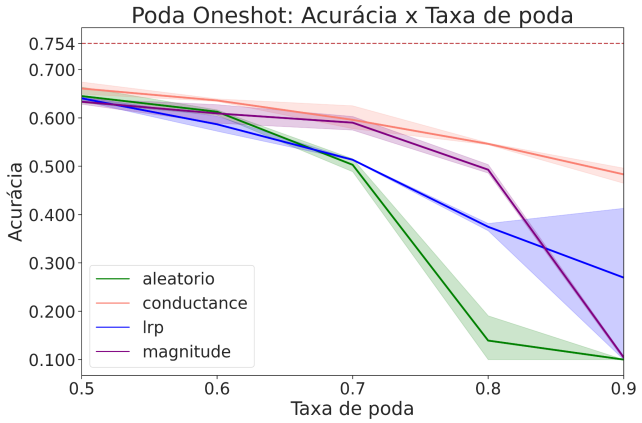

Figura 4: Gráfico de Acurácia x Taxa de Poda para os diferentes métodos de filtragem na poda *oneshot*

É importante ressaltar que, para percentuais de poda menos agressivos, os métodos demonstraram comportamento bastante similar, inclusive a poda aleatória. A Tabela 1 apresenta os resultados detalhados para cada cenário, fornecendo os desvios-padrão das acurácias de cada método de poda.

Método	0.5	0.6	0.7	0.8	0.9
conductance	**0.66 ± 0.01**	**0.64 ± 0.00**	**0.60 ± 0.03**	**0.55 ± 0.00**	**0.48 ± 0.02**
lrp	0.64 ± 0.00	0.59 ± 0.02	0.51 ± 0.00	0.37 ± 0.01	0.27 ± 0.16
magnitude	0.63 ± 0.01	0.61 ± 0.02	0.59 ± 0.01	0.49 ± 0.01	0.10 ± 0.00
aleatorio	0.65 ± 0.02	0.61 ± 0.01	0.50 ± 0.01	0.14 ± 0.05	0.10 ± 0.00

Tabela 1: Acurácia e desvio padrão dos diferentes métodos de filtragem na poda *oneshot*

4.3 Poda iterativa

De forma geral, os experimentos de poda iterativa mostraram resultados superiores em relação à poda *oneshot*, conforme ilustrado na Figura 9. A filtragem baseada em *conductance* continuou apresentando resultados superiores em relação aos demais métodos, mesmo em percentuais de poda mais agressivos. No entanto, a queda na acurácia de todos os métodos na taxa de poda de 0.9 foi mais suave, e desta vez a poda por magnitude superou a poda por LRP em termos de variância de resultados.

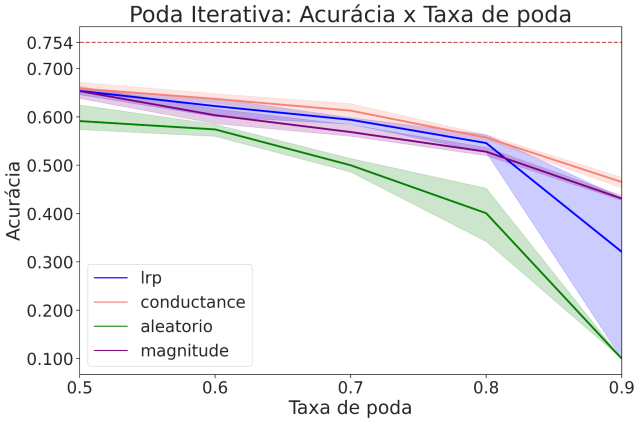

Figura 9: Gráfico de Acurácia x Taxa de Poda para os diferentes métodos de filtragem na poda iterativa

Uma vez mais, todos os métodos demonstraram resultados altamente similares em baixos percentuais de poda, como evidenciado na Tabela 2.

Método	0.5	0.6	0.7	0.8	0.9
conductance	**0.66 ± 0.01**	**0.64 ± 0.01**	**0.61 ± 0.02**	**0.56 ± 0.01**	**0.47 ± 0.01**
lrp	0.65 ± 0.01	0.62 ± 0.02	0.59 ± 0.01	0.55 ± 0.02	0.32 ± 0.19
magnitude	0.65 ± 0.01	0.60 ± 0.02	0.57 ± 0.01	0.53 ± 0.01	0.43 ± 0.01
aleatorio	0.59 ± 0.03	0.57 ± 0.01	0.50 ± 0.01	0.40 ± 0.06	0.10 ± 0.00

Tabela 2: Acurácia e desvio padrão dos diferentes métodos de filtragem na poda iterativa

5. CONCLUSÃO

Com base nos resultados apresentados, foi possível estabelecer uma relação entre o uso de métodos de explicabilidade como critérios para a poda de redes neurais profundas e melhorias na qualidade, consistência e magnitude da poda. Essa relação foi particularmente evidente no caso do método *conductance*, que demonstrou consistentemente resultados superiores em comparação aos demais métodos.

Além disso, os resultados também indicam que, para podas menos agressivas, a estratégia de filtragem utilizada pode não ter um impacto significativo. Nesses casos, é recomendável optar por métodos de filtragem mais simples, que não exijam etapas adicionais para a geração dos valores de importância.

De maneira geral, os resultados obtidos neste estudo são promissores e indicam que a utilização de métodos de explicabilidade como critério de poda em redes neurais profundas pode desempenhar um papel significativo na facilitação da operacionalização e na redução dos custos de infraestrutura associados aos modelos de *Deep Learning*. Essa abordagem permite a redução do tamanho dos modelos, mantendo níveis de acurácia comparáveis ou até superiores, o que contribui para a otimização do processamento de dados e a aceleração do treinamento. Além disso, ao reduzir o tamanho dos modelos, há uma diminuição do consumo de recursos computacionais em tempo de inferência, resultando em menor consumo energético e, consequentemente, na redução dos impactos ambientais relacionados à operação dos modelos em larga escala. Esses resultados indicam um caminho promissor para tornar os modelos de *Deep Learning* mais eficientes e sustentáveis, alinhando-se às demandas atuais por soluções tecnológicas mais responsáveis e de menor impacto ambiental.

Em relação a trabalhos futuros, sugere-se a realização de experimentos com arquiteturas mais profundas, como a VGG16 e VGG19 [18]. Além disso, o uso de conjuntos de dados mais desafiadores, como CIFAR-100 e ImageNet [2,10], poderia fornecer insights adicionais sobre a eficácia dos métodos de poda em cenários mais complexos.

Outra direção promissora para pesquisas futuras seria explorar outros métodos de filtragem explicativa, como DeepLift e SHAP [11,17]. Esses métodos têm o potencial de fornecer uma compreensão mais detalhada das características e atributos relevantes das redes neurais, contribuindo para uma poda mais eficiente e eficaz.

Essas investigações adicionais têm o potencial de alavancar a compreensão dos efeitos da poda em redes neurais profundas e expandir o conhecimento sobre o uso de métodos de explicabilidade nesse contexto.

6. AGRADECIMENTOS

Aos meus amados pais e familiares, expresso minha profunda gratidão pelo apoio incansável, tanto emocional quanto financeiro, que me concederam durante toda esta jornada. Vocês foram a luz que iluminou meu caminho, o alicerce que me sustentou nos momentos de desafio e a fonte inesgotável de amor e encorajamento. Sem vocês, nada disso seria possível.

Aos meus dedicados professores, em especial ao professor Campelo, sou imensamente grato pelas oportunidades únicas que me proporcionaram ao longo da graduação. Suas aulas foram verdadeiros faróis, guiando-me pelos intricados caminhos do conhecimento. Agradeço também ao meu estimado orientador, Herman, cuja paixão contagiante e sabedoria inspiradora despertaram meu interesse e me conduziram com cuidado e paciência ao longo deste trabalho.

Aos amigos que fiz ao longo dessa jornada acadêmica, a vocês dedico meu carinho e agradecimento. Vocês transformaram cada desafio em uma oportunidade para rir, aprender e crescer juntos.

Suas presenças calorosas e momentos compartilhados tornaram minha experiência mais leve, divertida e memorável.

A todos que contribuíram, de uma forma ou de outra, para a minha trajetória acadêmica, saibam que cada gesto de apoio, cada palavra de incentivo e cada momento de companheirismo foram fundamentais para a minha formação como estudante e como pessoa. Sou profundamente grato por ter atravessado esse percurso rodeado por pessoas tão especiais.

Que este agradecimento possa expressar a imensidão dos meus sentimentos de gratidão. Vocês são a essência que deu vida a cada página deste trabalho e o combustível que alimentou minha busca pelo conhecimento.

7. REFERÊNCIAS

[1] Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS One, 10(7), e0130140. doi:10.1371/journal.pone.0130140

[2] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition, 248–255. IEEE.

[3] Dhamdhere, K., Sundararajan, M., & Yan, Q. (2018). How important is a neuron? Retrieved from http://arxiv.org/abs/1805.12233

[4] Doshi-Velez, F., & Kim, B. (2017). Towards A rigorous science of interpretable machine learning. Retrieved from http://arxiv.org/abs/1702.08608

[5] Frankle, J., & Carbin, M. (2018). The lottery ticket hypothesis: Finding sparse, trainable neural networks. Retrieved from http://arxiv.org/abs/1803.03635

[6] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2019). A survey of methods for explaining black box models. ACM Computing Surveys, 51(5), 1–42. doi:10.1145/3236009

[7] Han, S., Pool, J., Tran, J., & Dally, W. J. (2015). Learning both weights and connections for efficient neural networks. Retrieved from http://arxiv.org/abs/1506.02626

[8] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. Retrieved from http://arxiv.org/abs/1512.03385

[9] Interpretable Machine Learning - Molnar, C. (2019). Interpretable Machine Learning: A Guide for Making Black Box Models Explainable.

[10] Krizhevsky, A. (n.d.). Learning multiple layers of features from tiny images. Retrieved June 14, 2023, from Toronto.edu website: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[11] Lundberg, S., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. Retrieved from http://arxiv.org/abs/1705.07874

[12] Magalhães, W. F. (2021). Poda estruturada de redes neurais convolucionais e a hipótese do bilhete de loteria. Retrieved from http://dspace.sti.ufcg.edu.br:8080/jspui/handle/riufcg/25035

[13] Molchanov, P., Tyree, S., Karras, T., Aila, T., & Kautz, J. (2016). Pruning convolutional neural networks for resource efficient inference. Retrieved from http://arxiv.org/abs/1611.06440

[14] Renda, A., Frankle, J., & Carbin, M. (2020). Comparing rewinding and fine-tuning in neural network pruning. Retrieved from http://arxiv.org/abs/2003.02389

[15] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. Retrieved from http://arxiv.org/abs/1801.04381

[16] Schwartz, R., Dodge, J., Smith, N. A., & Etzioni, O. (2019). Green AI. Retrieved from http://arxiv.org/abs/1907.10597

[17] Shrikumar, A., Greenside, P., & Kundaje, A. (2017). Learning important features through propagating activation differences. Retrieved from http://arxiv.org/abs/1704.02685

[18] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. Retrieved from http://arxiv.org/abs/1409.1556

[19] Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. Retrieved from http://arxiv.org/abs/1906.02243

[20] Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. Retrieved from http://arxiv.org/abs/1703.01365

[21] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is all you need. Retrieved from http://arxiv.org/abs/1706.03762

[22] Zhu, M. H., & Gupta, S. (2023). To prune, or not to prune: Exploring the efficacy of pruning for model compression. Retrieved from https://openreview.net/pdf?id=S1lN69AT-