

Detecção de Cães e Gatos: Um Estudo de Treinamento de Rede Neural para Classificação de Imagem

Campus Senador Helvídio Nunes de Barros - (UFPI)

Augusto Almondes Teixeira

Rafael Barbosa Costa

06 de fevereiro de 2024

Resumo

The article will cover the implementation of an object detection system using the Detectron2 library to identify cats and dogs in images. The process starts with data set preparation, including division into training, testing, and validation sets. Subsequently, the model's hyperparameters are configured by selecting the architecture and adjusting training settings such as maximum iterations and learning rate.

O artigo abordará a implementação de um sistema de detecção de objetos utilizando a biblioteca Detectron2 para identificar gatos e cachorros em imagens. O processo começa com a preparação dos conjuntos de dados, incluindo divisão em conjuntos de treinamento, teste e validação. Em seguida, são configurados os hiperparâmetros do modelo, escolhendo a arquitetura e ajustando as configurações de treinamento, como número máximo de iterações e taxa de aprendizado.

1 Introdução

A detecção de objetos, uma tarefa fundamental em visão computacional, desempenha um papel crucial em várias aplicações do mundo real, desde direção autônoma até sistemas de vigilância. Detectar e localizar objetos dentro de uma imagem fornecem informações valiosas para entender o conteúdo visual e possibilitar processos de tomada de decisão inteligentes. Entre os diversos frameworks de detecção de objetos disponíveis, o Detectron2, desenvolvido pelo Facebook AI Research (FAIR), se destaca como uma plataforma poderosa e flexível para a construção de sistemas de detecção de objetos de última geração.

Este artigo explora a implementação de um sistema de detecção de objetos utilizando a biblioteca Detectron2, com o objetivo específico de identificar dois animais domésticos populares: gatos e cachorros. A capacidade de reconhecer animais de estimação em imagens não é apenas de interesse para os proprietários de animais de estimação, mas também tem aplicações práticas em áreas como medicina veterinária, bem-estar animal e produtos relacionados a animais de estimação.

2 Metodologia

A metodologia adotada neste projeto consistiu em uma abordagem sistemática e cuidadosa para desenvolver e avaliar um sistema de detecção de objetos capaz de identificar gatos e cachorros em imagens. Inicialmente, foi realizada uma extensa pesquisa para selecionar um conjunto de dados adequado, composto por imagens representativas de gatos e cachorros. Para isso, a base de imagens foi obtida do site Roboflow, que oferece uma coleção diversificada de imagens de animais de estimação dentre outros diversos datasets com marcações precisas para as classes de gatos e cachorros.

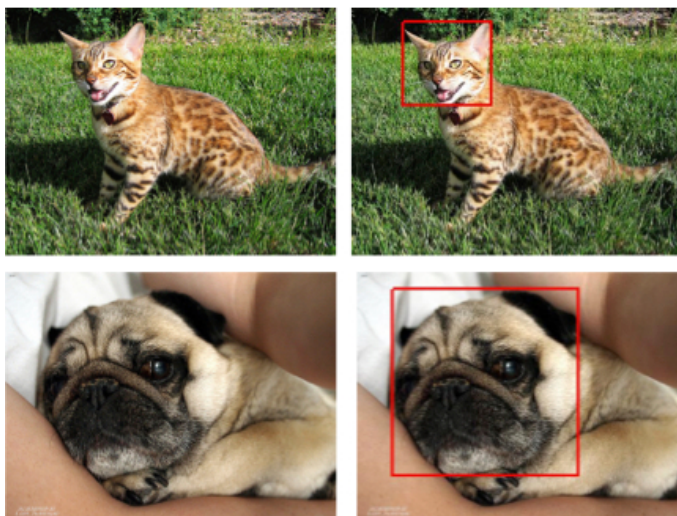


Figura 1: Figuras do site Roboflow

Após a aquisição do conjunto de dados, procedeu-se com sua preparação, que incluiu a organização das imagens em diretórios separados para cada classe, ou seja, gatos e cachorros. Além disso, cada imagem já veio com as marcações precisas para as localizações dos animais presentes na cena, tanto para os gatos quanto para os cachorros. Essas marcações foram fornecidas junto com o conjunto de dados e garantiram a qualidade e a precisão das informações desde o início do projeto.

Categoria/Testes	Instâncias
Cat	110
Dog	250

Tabela 1: Tabela de teste.

Categoria/Treino	Instâncias
Cat	1418
Dog	2991

Tabela 2: Tabela de treino.

2.1 Seleção da Arquitetura do Modelo e Treinamento

A arquitetura escolhida foi a Faster R-CNN como base para o modelo de detecção de objetos. Essa escolha foi feita com base na eficácia comprovada dessa arquitetura em tarefas de detecção de objetos e na disponibilidade de modelos pré-treinados no Detectron2 para essa arquitetura.

Antes de iniciar o treinamento do modelo de detecção de objetos, é essencial realizar a divisão do conjunto de dados em imagens de teste, treinamento e validação. Esta etapa é fundamental para avaliar o desempenho do modelo em dados não vistos durante o treinamento. Felizmente, o conjunto de dados obtido do site Roboflow já inclui essa divisão, facilitando o processo de preparação dos dados.

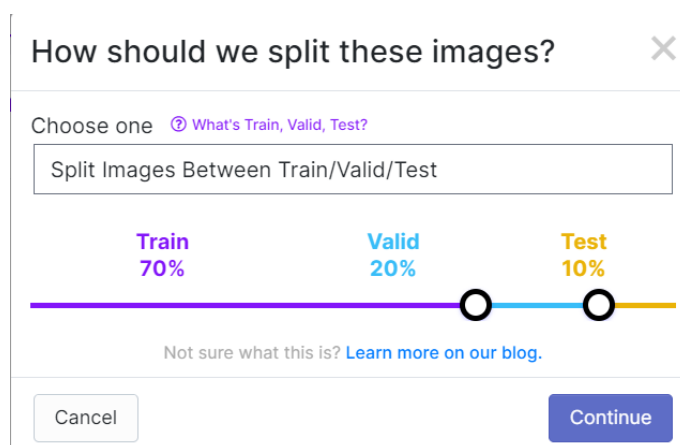


Figura 2: Imagem divisão do conjunto de dados.

Durante a fase inicial do treinamento do modelo, foram observados resultados limitados devido a uma configuração inadequada dos hiperparâmetros. Inicialmente, o número de iterações foi definido como relativamente baixo e apenas duas classes foram consideradas. Como resultado, o modelo foi capaz de reconhecer apenas gatos em algumas imagens, enquanto outras não foram identificadas corretamente.

Na segunda iteração do teste, apesar do aumento no número de iterações, o modelo ainda falhou em detectar a classe de cachorros. Isso foi atribuído à persistência da configuração de apenas duas classes, o que limitou a capacidade do modelo em reconhecer objetos além dos gatos.

Na terceira iteração do teste, foi mantido o número de iterações ajustado na iteração anterior, porém o número de classes foi aumentado para três. Embora apenas um gato e um cachorro fossem objetos de interesse, foi necessário incluir uma terceira classe para representar o "background". Essa alteração permitiu que o modelo reconhecesse tanto gatos quanto cachorros com sucesso. Essas iterações no treinamento do modelo destacam a importância da configuração cuidadosa dos hiperparâmetros e da consideração adequada do número de classes durante o desenvolvimento de sistemas de detecção de objetos.

2.2 Resultados

Durante a etapa de avaliação do modelo treinado, uma série de testes foi conduzida para avaliar sua capacidade de reconhecer objetos individualmente e em conjunto em diferentes cenários. Os testes foram organizados da seguinte forma:

Teste de Reconhecimento de Gato Isolado: Neste teste, o modelo foi submetido a imagens contendo apenas um gato. O objetivo era verificar se o modelo era capaz de identificar corretamente a presença e a localização do gato na imagem.



Figura 3: Teste do treinamento em imagem de gato

Teste de Reconhecimento de Cachorro Isolado: Similarmente ao teste anterior, imagens contendo apenas um cachorro foram utilizadas para avaliar a capacidade do modelo em detectar e delimitar a presença do cachorro na cena.

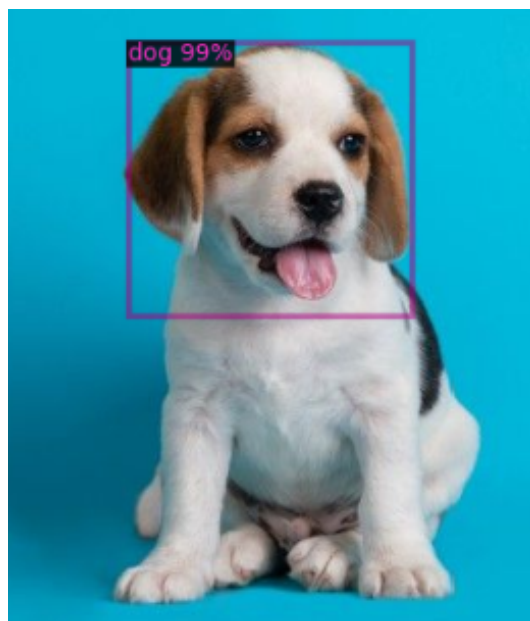


Figura 4: Teste do treinamento em imagem de cachorro

Teste de Reconhecimento de Gato e Cachorro em Conjunto: Neste teste, imagens contendo tanto um gato quanto um cachorro foram apresentadas ao modelo. O objetivo era verificar se o modelo poderia identificar simultaneamente a presença e as localizações de ambos os animais na mesma imagem.

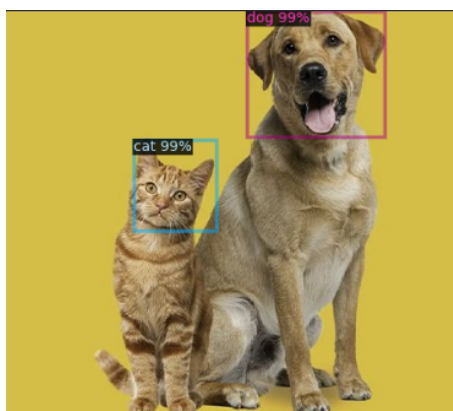


Figura 5: Teste do treinamento em imagem de um gato e cachorro

Cada teste foi projetado para representar cenários realistas nos quais o modelo seria empregado, permitindo uma avaliação abrangente de sua eficácia e robustez. Os resultados desses testes forneceram insights valiosos sobre o desempenho do modelo em diferentes contextos e foram fundamentais para validar sua capacidade de detecção de objetos em imagens contendo gatos e cachorros.

Após o treinamento, foram utilizadas ferramentas de visualização, como o TensorBoard, para realizar a plotagem dos gráficos e buscar os dados relevantes para análise.

A "total-loss" é uma métrica comumente usada em treinamentos de modelos de aprendizado profundo, incluindo modelos de detecção de objetos como a Faster R-CNN. Ela representa a soma de todas as perdas individuais calculadas durante o treinamento do modelo.

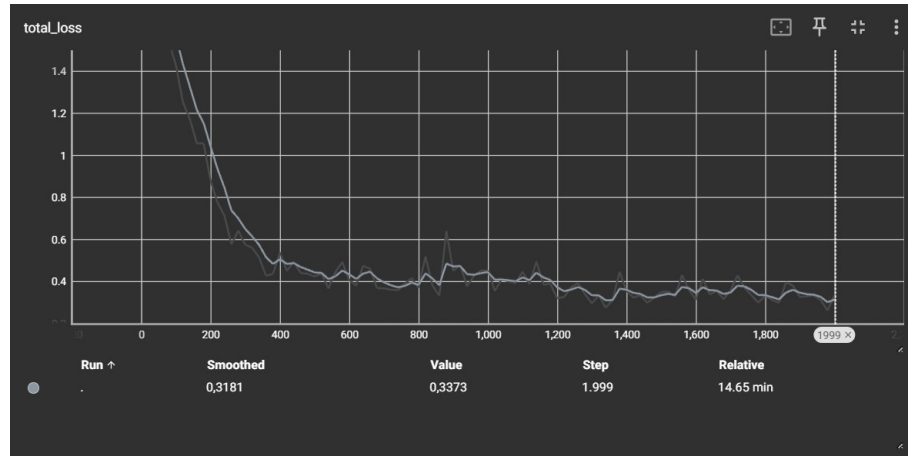


Figura 6: Imagem total-loss

"False-negative" (falso negativo) é um termo comumente utilizado em problemas de classificação e detecção de objetos para descrever uma situação em que o modelo prevê erroneamente a ausência de um objeto que está presente na verdade. Em outras palavras, um falso negativo ocorre quando o modelo falha na detecção de um objeto que deveria ter sido identificado de maneira correta.

No contexto de detecção de objetos, os falsos negativos são especialmente preocupantes, pois podem levar a consequências graves em aplicações práticas.

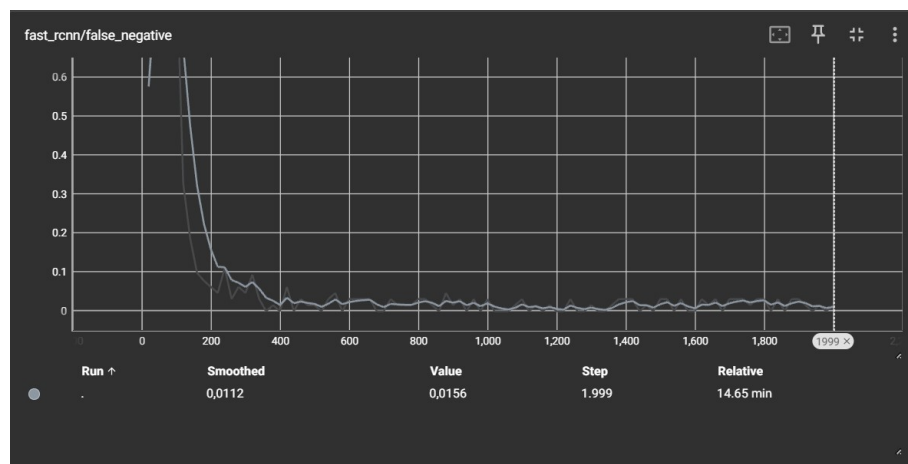


Figura 7: Imagem false-negative

"Cls-accuracy" (precisão de classe) é uma métrica utilizada para avaliar o desempenho de um modelo de classificação em relação a cada classe individualmente. Essa métrica mede a

proporção de previsões corretas feitas pelo modelo para uma classe específica em relação ao número total de previsões feitas para essa classe.

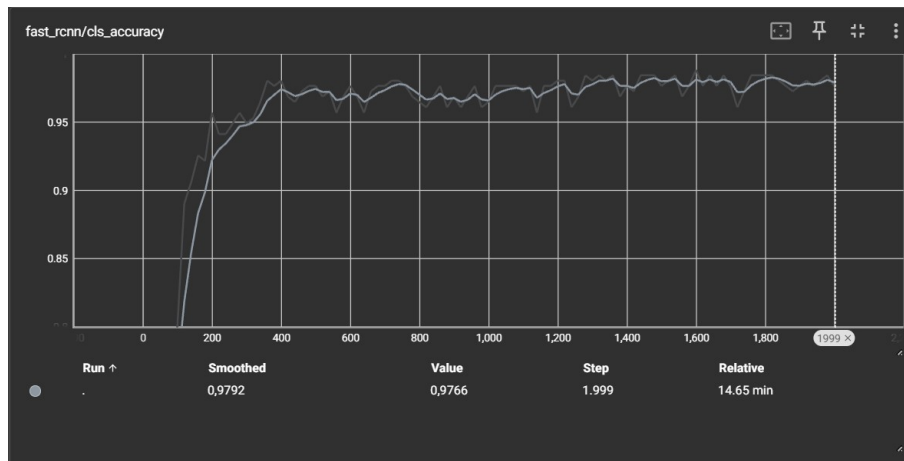


Figura 8: Imagem cls-accuracy

3 Discussão

A discussão dos resultados obtidos no desenvolvimento do método de detecção de objetos envolve uma análise cuidadosa dos processos e das decisões tomadas durante o projeto. Os resultados alcançados refletem uma combinação de fatores, incluindo a escolha da arquitetura do modelo, os ajustes dos hiperparâmetros e a qualidade do conjunto de dados utilizado.

Os resultados limitados observados durante as primeiras iterações do treinamento destacaram a importância da configuração adequada dos hiperparâmetros, como o número de iterações e o número de classes. A persistência de configurações inadequadas resultou em dificuldades na detecção de objetos além das classes inicialmente consideradas, evidenciando a necessidade de ajustes e refinamentos.

Além disso, as limitações do trabalho desenvolvido também precisam ser consideradas. Uma delas é a dependência da qualidade e da representatividade do conjunto de dados utilizado. Mesmo com a divisão prévia do conjunto de dados, a presença de dados desbalanceados ou de baixa qualidade pode afetar negativamente o desempenho do modelo.

Outra limitação importante está relacionada à complexidade das cenas e à diversidade dos objetos presentes nas imagens. O modelo pode enfrentar dificuldades na detecção de objetos em cenários desafiadores, como condições de iluminação adversas, sobreposição de objetos e variações na escala e na orientação dos objetos.

Em resumo, a discussão dos resultados destaca a importância de uma abordagem criteriosa no desenvolvimento de métodos de detecção de objetos, enfatizando a necessidade de ajustes finos dos hiperparâmetros, a qualidade do conjunto de dados e a consideração das limitações

inerentes ao problema. Essa análise crítica dos resultados é fundamental para identificar áreas de melhoria e orientar futuras pesquisas na área de visão computacional.

4 Conclusão

A conclusão do estudo destaca o desenvolvimento de um método de detecção de objetos com base na arquitetura Faster R-CNN para identificação de gatos e cachorros em imagens. A superação de desafios iniciais, como configurações inadequadas de hiperparâmetros, foi crucial para alcançar resultados promissores. A divisão prévia do conjunto de dados facilitou a preparação dos dados e proporcionou uma base sólida para o treinamento do modelo. Os resultados obtidos contribuem significativamente para o avanço da pesquisa em visão computacional, fornecendo insights valiosos e estabelecendo um ponto de partida para futuras melhorias e aplicações práticas.