

Relatório do Projeto de Detecção de Doenças em Cafeeiros

Rodrigo Marques Duarte
João Pedro Alves Carneiro Valadão
Lucas da Silva Rosa
Kaique Inácio Salvador

1. Introdução

A cafeicultura é uma atividade econômica de grande importância, mas enfrenta desafios relacionados ao surgimento de doenças que afetam a produtividade e a qualidade dos grãos. Para mitigar esses impactos, este projeto propõe o desenvolvimento de um sistema inteligente capaz de identificar e classificar doenças em folhas de cafeeiros a partir de imagens. A solução combina visão computacional e inteligência artificial, utilizando um modelo de deep learning baseado em YOLO para a detecção visual das doenças e um modelo de linguagem para fornecer descrições detalhadas e recomendações de tratamento. O objetivo é oferecer uma ferramenta acessível e eficiente para auxiliar produtores na detecção precoce de problemas fitossanitários, promovendo uma gestão mais eficaz das lavouras.

2. Tecnologias Utilizadas

1. **Backend:** Flask (API RESTful)
2. **Frontend:** HTML, CSS, JavaScript (Express.js para servir a aplicação web)
3. **Banco de Dados:** Não aplicável neste momento
4. **Machine Learning:** YOLO (Ultralytics) para classificação das doenças
5. **Processamento de Linguagem Natural:** Modelo LLaMA (via Ollama API)
6. **Containerização:** Docker
7. **Bibliotecas:** OpenCV, NumPy, Pandas, Flask-CORS, Requests, Transformes, Torch

3. Estrutura do Projeto

O projeto é dividido em:

1. **API Flask:** Contém dois serviços principais:
 - **servico1:** Recebe uma imagem, processa com YOLO e retorna a previsão de doença e a descrição via LLaMA.
 - **servico2:** Recebe um texto do usuário e retorna uma resposta processada pelo modelo de linguagem.
2. **Frontend Web:** Formulário para upload de imagens e texto, exibindo as respostas do modelo.

3. **Modelo de Machine Learning:**
 - Treinado usando imagens de folhas de cafeeiros, classifica doenças como **ferrugem** e **bicho mineiro**.
 - Implementado com **YOLOv8** para alta precisão e rapidez na inferência.
4. **Docker:** Arquivos Dockerfile para implantação do sistema em ambiente isolado.

4. Processamento da Imagem e Inferência

1. O usuário faz upload de uma imagem.
2. A API Flask salva a imagem temporariamente.
3. O modelo YOLO processa a imagem e retorna:
 - Classe da doença detectada
 - Confiança da previsão
4. O modelo de linguagem gera uma descrição e sugestão de tratamento.
5. A resposta é enviada ao front-end.

5. Treinamento do Modelo

1. Utiliza dataset dividido em **train** e **val** (80% / 20%).
2. Treinamento por 5 épocas com batch size 8 e imagens de 640x640.
3. Modelo salvo como **coffee_disease_model.pt**.

6. Resultados

Os testes mostram boa precisão do modelo na detecção de **ferrugem** e **bicho mineiro**, com confiança de 98%.

7. Melhorias Futuras

1. Adicionar mais classes de doenças ao modelo.
2. Melhorar a interface web.
3. Implementar uma base de dados para armazenar histórico de diagnósticos.
4. Melhorar a interação do modelo de linguagem para respostas mais detalhadas.

8. Conclusão

O sistema desenvolvido demonstra a viabilidade do uso de inteligência artificial na detecção de doenças em cafeeiros, proporcionando um fluxo eficiente para análise automatizada das folhas. A combinação de modelos de visão computacional e linguagem natural permite não apenas a identificação precisa das doenças, mas também a oferta de recomendações úteis aos produtores. Apesar dos avanços alcançados, melhorias futuras serão implementadas para aumentar a precisão do modelo, aprimorar a interface do usuário e expandir a base de dados, tornando a ferramenta ainda mais robusta e acessível para o setor agrícola.