

Machine Learning Project Documentation

format

1. Introduction

- **Project Title:** Pollen's Profiling: Automated Classification of Pollen Grains
- **Team Members:**
 1. Eduri Maryjones (Data collection,Train the model,Application building)
 2. Idimukkala Yasasswini (Train the model,save the model,Test the model,Application building)
 3. Mamidela Venkata Naga Suseel Kumar (Read the data (Exploratory Data Analysis) ,Image pre-processing,Training the model,Save the model,Test the model,Application building)

2. Project Overview

- **Purpose:** Automate pollen grain identification using a deep learning CNN model based on image classification.
- **Features:** Image upload for pollen classification.
Prediction of pollen type.
Web-based interface using Flask.

3. Architecture

- **Frontend:** Describe the frontend architecture using React.
- **Backend:** Outline the backend architecture using Node.js and Express.js.
- **Database:** Detail the database schema and interactions with MongoDB.

4. Setup Instructions

- **Prerequisites:** Python, TensorFlow, Flask, OpenCV, Scikit-learn, Numpy, Pandas, Matplotlib.
- **Installation:**

```
git clone https://github.com/YourUsername/Pollens-Profiling.git
cd Pollens-Profiling
pip install -r requirements.txt
```

5. Folder Structure

- **Client:** Describe the structure of the React frontend.
- **Server:** Explain the organization of the Node.js backend.

6. Running the Application

- Provide commands to start the frontend and backend servers locally.
 - **Frontend:** `npm start` in the client directory.
 - **Backend:** `npm start` in the server directory.

7. API Documentation

- Document all endpoints exposed by the backend.
- Include request methods, parameters, and example responses.

8. Authentication

- Explain how authentication and authorization are handled in the project.
- Include details about tokens, sessions, or any other methods used.

9. User Interface

- Provide screenshots or GIFs showcasing different UI features.

10. Testing

- Describe the testing strategy and tools used.

11. Screenshots or Demo

- Provide screenshots or a link to a demo to showcase the application.

12. Known Issues

- Accuracy depends on dataset quality.
- Misclassification in very similar pollen types.

13. Future Enhancements

- Deploy on cloud (Render, Heroku)
- Add user login to save predictions.
- Improve model with a larger dataset.