**Deep Learning**

**CT3 Mini Project**

**Inference Report**

**Dataset Loading:**

The dataset was successfully loaded using tensorflow\_datasets as ds\_train (training data) and ds\_test (testing data). The Beans dataset includes images of bean leaves categorized into three classes: angular\_leaf\_spot, bean\_rust, and healthy. The dataset is curated to support agricultural research and annotated by experts, enhancing its reliability.

* **Dataset Structure**: It contains 1034 training samples, 128 test samples, and 133 validation samples, with each image sized 500x500 pixels. The dataset format also includes labels for supervised learning, which helps the model identify the leaf disease type.

**Data Preprocessing:**

Each image was resized to 224x224 pixels, which matches the input requirements of MobileNet, and normalized to a range of [0, 1]. Resizing ensures compatibility with the MobileNet model, and normalization helps the model converge faster during training. Batching (32 images per batch) and prefetching were used to optimize data loading, especially when working with large datasets, ensuring smooth and efficient training.

**Data Visualization:**

A subset of images from the training set was visualized, showing distinct patterns across the classes. Observations included:

* **Clear Class Distinctions:** Each class had unique visual features. For instance, disease-infected leaves displayed spots or discoloration, while healthy leaves appeared uniformly green.
* **Potential Challenges:** Variations in lighting and leaf positioning were noted, which could impact model performance. This variation may necessitate further data augmentation in a production setting.

**Model Setup for Transfer Learning:**

We used MobileNet with weights pre-trained on the ImageNet dataset and applied transfer learning:

* **Base Model Freezing:** The initial layers were frozen, allowing the model to retain generalized features from ImageNet while learning specific disease characteristics from the bean dataset.
* **Additional Layers:** Custom classification layers were added, including a GlobalAveragePooling2D layer, followed by a dense layer with 128 neurons, and a final output layer using softmax activation to classify into three classes.
* **Rationale:** This setup helps leverage existing learned features from a robust model while adjusting the final layers for our dataset's requirements.

**Training the Model:**

The model was trained with early stopping, with a patience parameter set to 3 to prevent overfitting.

**Inference:**

* **Performance:** After 10 epochs, the model achieved high accuracy, with a gradual decrease in validation loss.
* **Training Parameters:** We used a batch size of 32, Adam optimizer with a default learning rate, and a maximum of 10 epochs. Training performance, as seen in the epoch outputs, improved quickly, reaching high accuracy and low validation loss.

**Model Evaluation:**

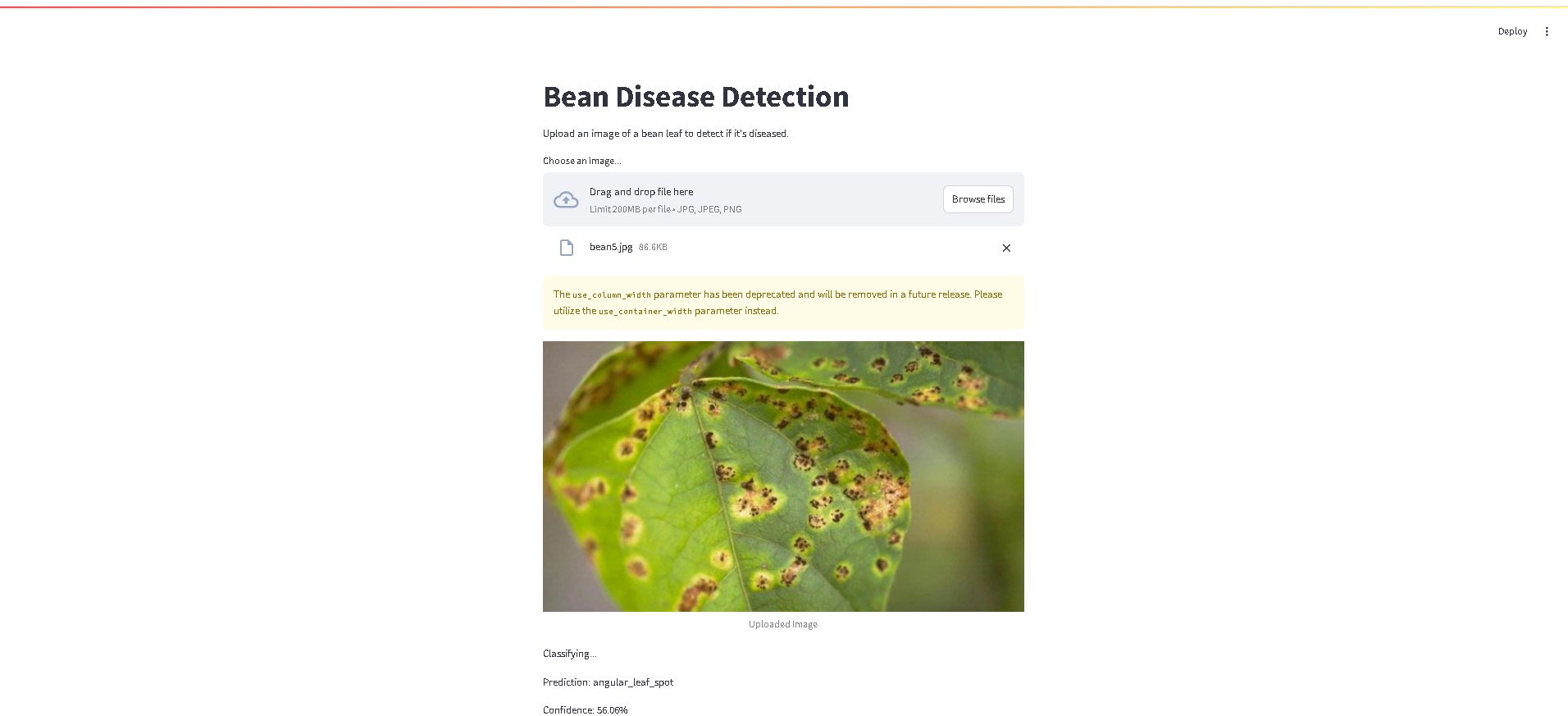
**Evaluation metrics on the test dataset are as follows:**

* **Accuracy:** 91%. **Precision, Recall, and F1-Score:** **Angular Leaf Spot:** Precision: 0.91, Recall: 0.91, F1-score: 0.91. **Bean Rust:** Precision: 0.86, Recall: 0.88, F1-score: 0.87. **Healthy:** Precision: 0.98, Recall: 0.95, F1-score: 0.96.

**Inference:**

* The model performed very well in distinguishing between healthy and diseased leaves, achieving over 90% accuracy and F1-scores close to 1.0 for each class.
* The highest precision and recall were observed for the healthy class, suggesting that the model finds healthy leaves easier to classify. Bean Rust presented a slightly lower score, indicating a need for potential data augmentation or fine-tuning specific to this class.

**Streamlit Application:**

****

**Inference:**

* Uploaded the bean image in the Streamlit application
* Prediction made is angular\_leaf\_spot.
* Confidence: 56.06%.