Project development
Model performance

Salesforce-Based AI Project Phases: Rotten Fruits & Vegetables Detection

---

🟢 1. Requirement Gathering & Use Case Finalization

Goal:

- Automate quality inspection by detecting rotten produce.
- Sync results into Salesforce for tracking and customer feedback.

Salesforce Role:

- Define objects like `Produce Inspection`, `Farmer`, `Vendor`, `Batch`.
- Identify where AI results will be stored and shown (e.g., in custom fields or dashboards).

---

2. Data Collection & Preparation

Data:

- Images of rotten and fresh fruits/vegetables.
- Source: Camera uploads, field agents, existing datasets**.**

Salesforce Integration:

- Use Salesforce Experience Cloud or mobile SDK for field data collection.
- Store metadata (e.g., batch ID, location, image URL) in Salesforce objects.

✅ *Model Performance Focus*:

- Clean, labeled, high-resolution images.
- Balanced dataset = better precision and recall.

🧪 3. Model Development (Outside Salesforce)

Tools: TensorFlow, Keras, PyTorch, Google AutoML, etc.

Model Types:

- CNN / Transfer Learning (MobileNet, ResNet)
- Binary classification: *Fresh* vs *Rotten*

Training:

- Augmentation, tuning hyperparameters, validation checks.

📊 Track Metrics:

Metric	Target
Accuracy	>90%
Recall (Rotten)	>95%

F1 Score	>92%

🔗 4. Model Integration with Salesforce

Approaches:

- Einstein Vision (Salesforce's native AI) — if using images.
- External Model API — deploy model to a cloud (e.g., AWS Lambda, GCP, Azure) and connect via Salesforce Apex callouts or MuleSoft**.**

Steps:

- Upload image from Salesforce UI or mobile app.
- Image is sent to the ML model.
- Result (fresh/rotten) is returned and stored in a Salesforce object.

✅ *Performance Considerations*:

- Response time < 2 sec
- Confidence threshold > 85%

📈 5. Prediction Results & Reporting

Inside Salesforce:

- Show result in a Lightning Component or Flow Screen.
- Use Reports & Dashboards for:

    - % Rotten by Batch
    - Vendor performance
    - Geographic trends

✅ *Performance Monitoring*:

- Track prediction vs human-verified outcomes.
- Use Einstein Discovery for further insights.

🔁 6. Continuous Learning & Feedback Loop

From Salesforce:

- Allow users to correct wrong predictions.
- Store feedback in a custom object (e.g., `Inspection Feedback`).
- Periodically export data and retrain the model.

✅ *Improves F1 Score & Generalization over time*

🔄 7. Automation with Flows & Alerts

- If `Rotten = True`, trigger:
    - Email to quality manager
    - Task creation for inspection team

- **Integration to inventory system to isolate bad produce**

🎯 Summary Table: Performance + Salesforce Actions

Phase	Model Performance Goal	Salesforce Integration
Data Collection	High-quality, diverse images	Mobile app data capture, object storage
Model Training	Accuracy > 90%, F1 > 92%	Trained externally
Model Inference	Fast, confident predictions	Apex callouts / Einstein Vision API
Results Display	Clear, actionable output	Lightning pages, record updates
Monitoring & Feedback	Improve model over time	Feedback loop via forms/flows
Automation & Reporting	Real-time action from results	Flows, reports, alerts, dashboards