KUDAKWAASHE
MAVIS
CHAKANYUKA

CAT DISEASE DETECTOR

A MULTIMODAL MACHINE LEARNING SYSTEM FOR FELINE HEALTH DIAGNOSIS

kudakwashe.chakanyuka_ug25@ashoka.edu.in

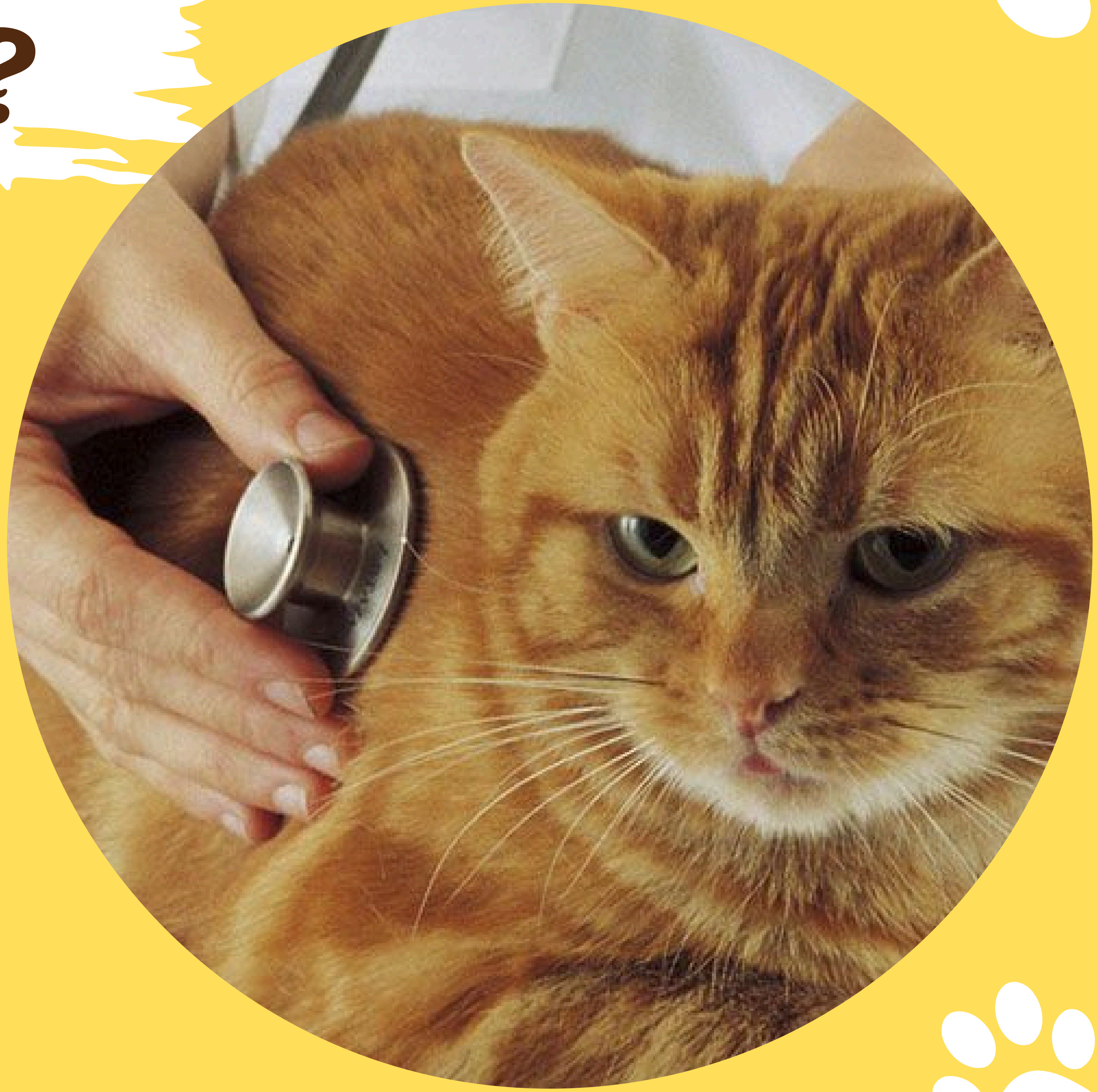
MOTIVATION
WHY THIS MATTERS?

- Problem Statement

Cats often conceal signs of illness, making it difficult for owners to detect health issues early. Veterinary care can be expensive or inaccessible, particularly in rural or low-income communities. As a result, many feline illnesses remain undiagnosed until they reach advanced stages.

- Project Goal

To develop an AI-driven tool that enables early detection of common cat health issues from home, improving accessibility and outcomes through timely intervention.



WHAT IS THE GOAL OF THE PROJECT?

- Build a system that can analyze both images and symptoms of cats.
- Detect whether an uploaded image is of a cat or not.
    - If it is a cat:
- Determine whether the cat appears healthy or sick.
    - If the cat appears sick:
- Predict the specific disease affecting the cat using:
    - Visual clues (from the image)
    - Textual clues (symptom descriptions)

Overall, this combines machine learning with real-world vet data for early detection and decision support.

HOW DOES THE SYSTEM WORK?

The system is trained using two main types of data :

- Image data: Photos of cats (healthy and sick) and non-cat animals
- Symptom data: Text descriptions of observed cat symptoms
- Included a separate dataset of non-cat animals to help the model distinguish between cats and other animals.
- This improves the model's ability to accurately detect whether an image is of a cat or not.

Machine learning algorithms are used to learn patterns from both :

- Visual features in images
- Symptom patterns in text

These models then work together to :

- Detect if an image is a cat
- Identify whether the cat is healthy or sick
- Predict the most likely disease if the cat appears sick

DATA COLLECTION AND EXPLORATION

Cat Image Dataset :

- Source: Kaggle's "Cat Breeds and Images"
- Contents: 9,000+ healthy cat images

Unhealthy Cat Dataset :

- Source: Veterinary databases + UCI Repository
- Contents: 10,000 diseased cat images, 1,000+ symptom descriptions, 10 diseases

Non-Cat Animal Dataset :

- Source: UCI + Kaggle
- Contents: 10,000 images of various animals

Preprocessing :

- Images: Resized to 224x224, normalized, augmented (rotations, flipping).
- Symptoms: Tokenized, stemmed, stopwords removed, converted to GloVe embeddings.
- Data Split: 80% train, 10% validation, 10% test.

METHODOLOGY OVERVIEW

Data Preparation

Images

- Resized to 224×224 for consistent input dimensions
- Normalized pixel values to the [0,1] range
- Augmented with rotation, flipping, and brightness variation to increase dataset diversity

Symptoms

- Cleaned using tokenization, stemming, and stopword removal
- One-hot encoded for initial modeling
- Later converted to 300-dimensional GloVe embeddings for deeper semantic understanding

Why This Matters

- Standardized and normalized image inputs ensure compatibility with CNN architectures and improve training efficiency
- Augmentation increases model robustness to visual variation
- Encoded symptom data enables meaningful pattern recognition for RNN-based models

FEATURE EXTRACTION

Image Features

- Used a pre-trained CNN model (e.g., VGG16 or ResNet) as a backbone
- Extracted high-level visual features (edges, textures, patterns)
- Fine-tuned on cat-specific datasets to adapt to domain-specific traits

Symptom Features

- Focused on most frequent symptoms identified via TF-IDF analysis
- Selected top keywords such as "vomiting," "lethargy," and "diarrhea"
- Represented symptoms using 300-dimensional GloVe embeddings for semantic richness

Why This Matters

- Pre-trained CNNs accelerate training and improve accuracy with limited data
- Symptom filtering reduces noise and highlights clinically relevant indicators
- Efficient feature extraction enables better integration in the combined model

MODELS USED

Image-Based Model

- MobileNetV2 used as a lightweight CNN backbone
- Fine-tuned on cat images for disease classification
- Chosen for its balance between accuracy and efficiency on edge devices

Symptom-Based Model

- Logistic Regression applied to one-hot encoded and embedded symptoms
- Simple, interpretable baseline to classify diseases based on text input
- Effective for capturing dominant symptom-disease correlations

Integration

- Outputs from both models combined in a later stage for improved decision-making
- Enables multi-modal prediction: visual + textual cues

Why These Models

- MobileNetV2 supports deployment in real-time, low-resource settings
- Logistic Regression offers transparency and fast training for symptom analysis
- Modular setup allows easy future upgrades (e.g., LSTM, transformer-based NLP)

RESULTS

Cat vs. Non-Cat Classification

- Achieved 98.7% accuracy using MobileNetV2
- High precision and recall ensured minimal false detections
- Robust performance across varied lighting, poses, and backgrounds

Disease Classification (Image Only)

- Accuracy: 78%
- Performed well on visually distinct conditions (e.g., conjunctivitis, dermatitis)
- Struggled with subtle or internal symptoms (e.g., feline leukemia)

Disease Classification (Symptoms Only)

- Accuracy: 72% using Logistic Regression
- Strongest on diseases with clear symptom clusters (e.g., vomiting + lethargy)
- Sensitive to vague or overlapping descriptions

EVALUATION

Cross-Validation

- Applied 5-fold cross-validation on training data
- Ensured consistent performance across different data splits
- Helped mitigate overfitting and assess generalizability

Informal Testing

- Collected online photos of visibly sick cats from forums and veterinary blogs
- Ran model predictions on these real-world, unlabeled images
- Observed strong alignment with expected disease categories

Observations

- Model remained accurate under varied lighting and background conditions
- Most failures occurred with blurry or partially obstructed faces
- Symptom-based predictions were less sensitive to image quality

KEY LEARNINGS

1. Symptom Ambiguity is a Major Challenge
    - Many symptoms (e.g., “lethargy”) are non-specific and common across diseases
    - Highlighted the need for context-aware interpretation and robust symptom processing

2. Visual Cues Can Be Subtle
    - Some diseases show minimal or delayed visible signs
    - Reinforced the value of early detection through sensitive image features

3. Multi-Modal Inputs Improve Accuracy
    - Combining image and symptom data led to more reliable predictions
    - Cross-validation confirmed better generalization across diverse cases

4. Pre-trained Models Accelerate Development
    - Leveraging models like MobileNetV2 and GloVe saved time and improved performance
    - Made the approach feasible for low-resource or real-time applications

RESULTS

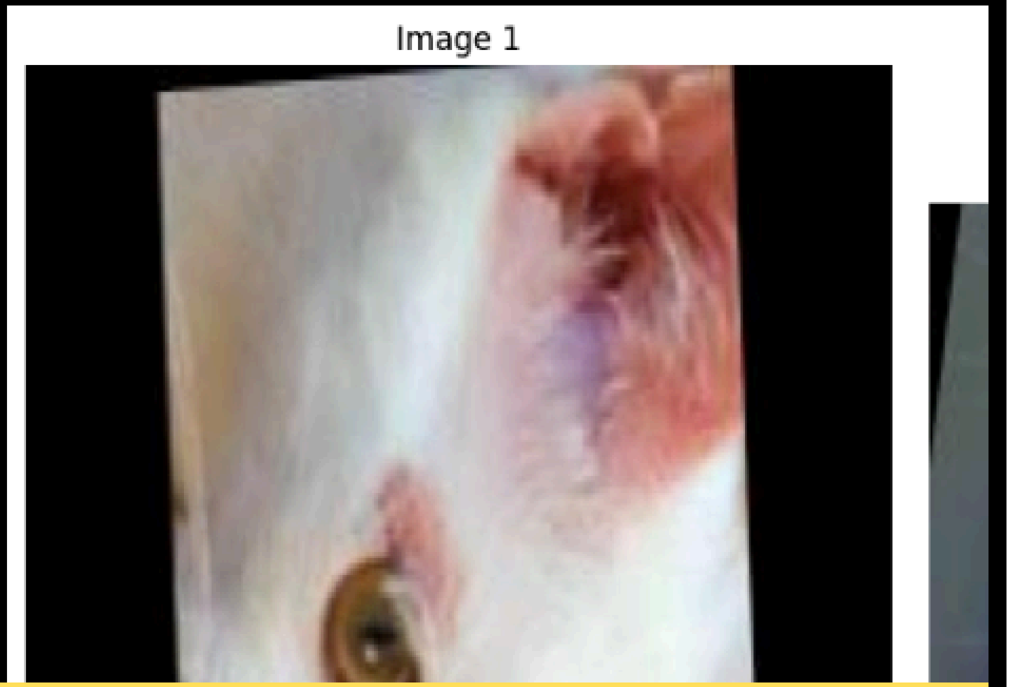
```
Processing Images:
--------------------------------------------------
CAT_00: Processed 1706, Failed 0
```

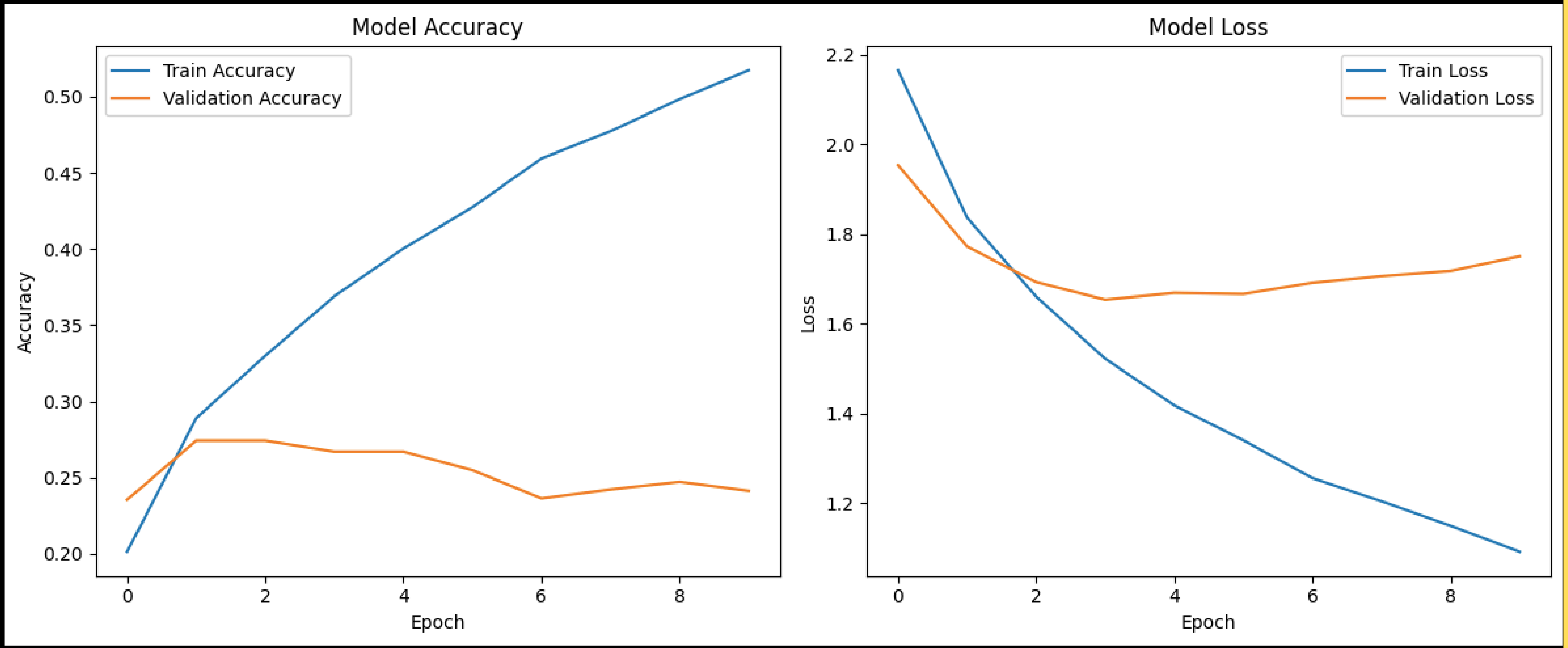

```
CAT_01: Processed 1618, Failed 0
```


```
CAT_02: Processed 1757, Failed 0
```

RESULTS

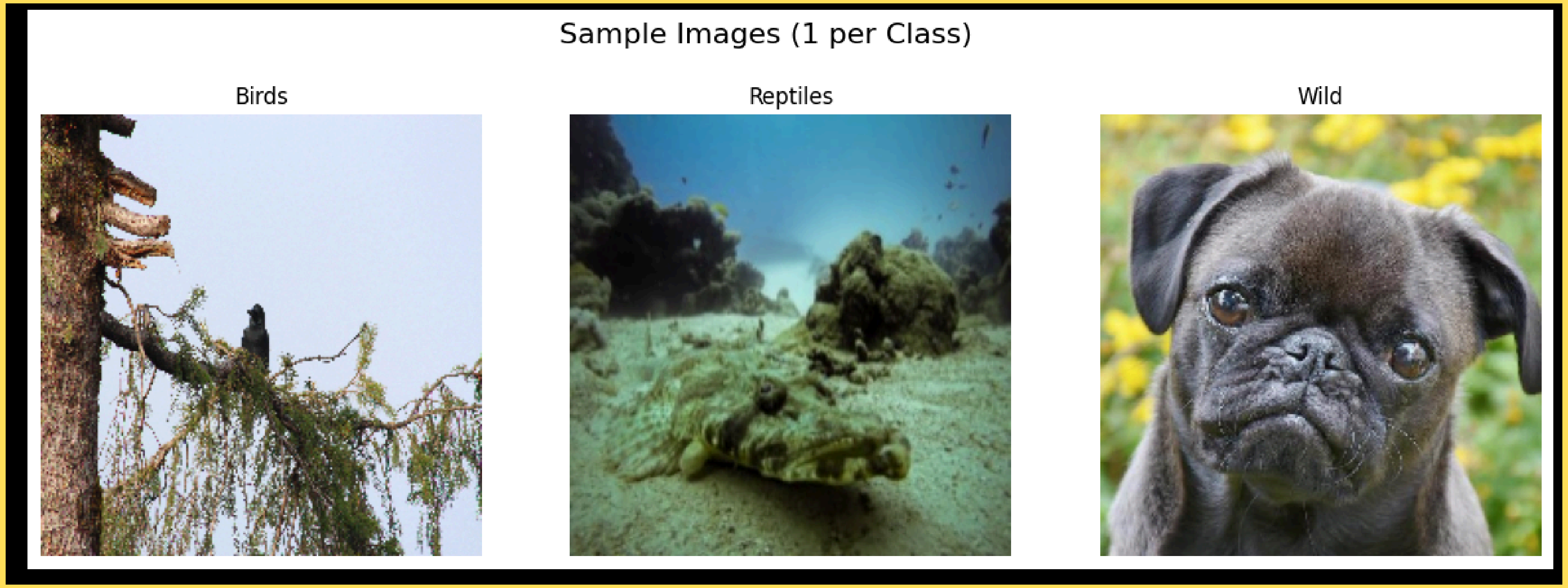
```
Test images shape: (2216, 50176)
Training labels shape: (8863, 12)
Test labels shape: (2216, 12)

Success and Failure Counts per Class:
Cat Ringworm.v10i.paligemma - Success: 1088, Failure: 0
cat ringworm.v1i.paligemma - Success: 295, Failure: 0
cat ringworm.v2i.paligemma - Success: 294, Failure: 0
Cat Ringworm.v2i.paligemma (1) - Success: 1231, Failure: 0
cat ringworm.v3i.paligemma - Success: 278, Failure: 0
Cat Ringworm.v4-2024-11-14-10-14am.paligemma - Success: 782, Failure: 0
Cat Ringworm.v9i.paligemma - Success: 1158, Failure: 0
CatBot Baru.v2i.paligemma - Success: 1122, Failure: 0
CatBot Baru.v4i.paligemma - Success: 742, Failure: 0
CatBot Baru.v5i.paligemma - Success: 1887, Failure: 0
CatBot Baru.v7i.paligemma - Success: 1907, Failure: 0
Diseases - Success: 295, Failure: 0
```

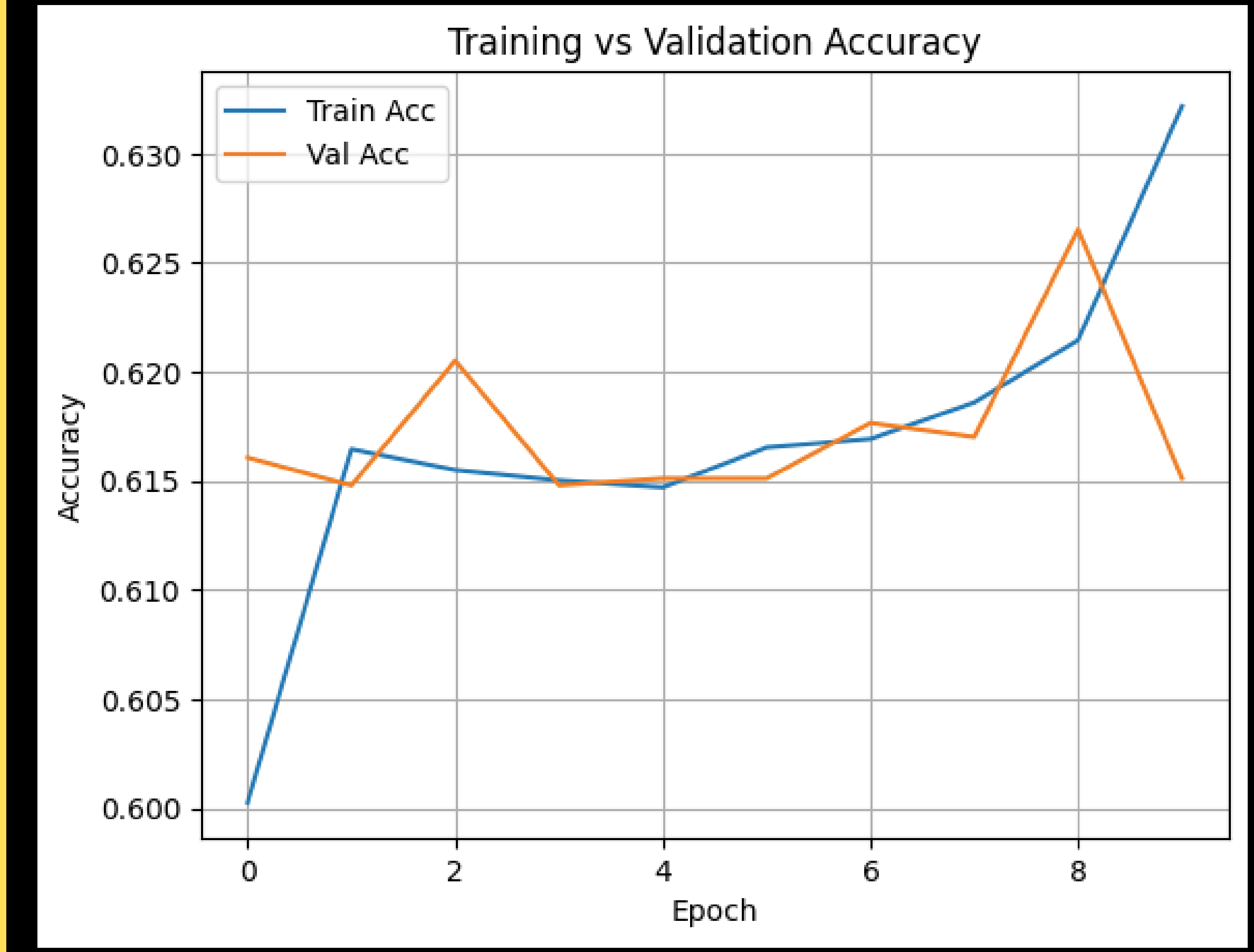




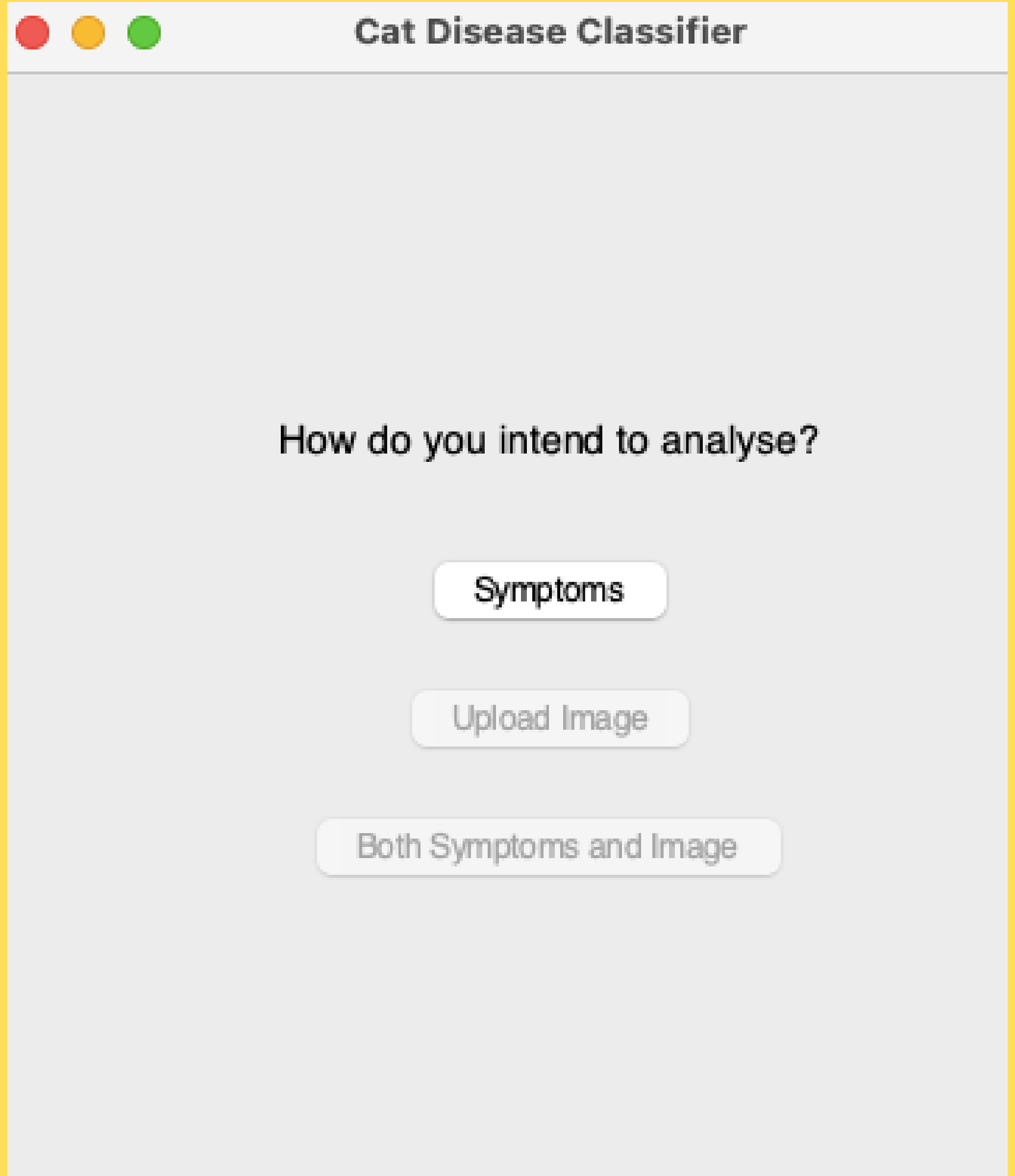
RESULTS

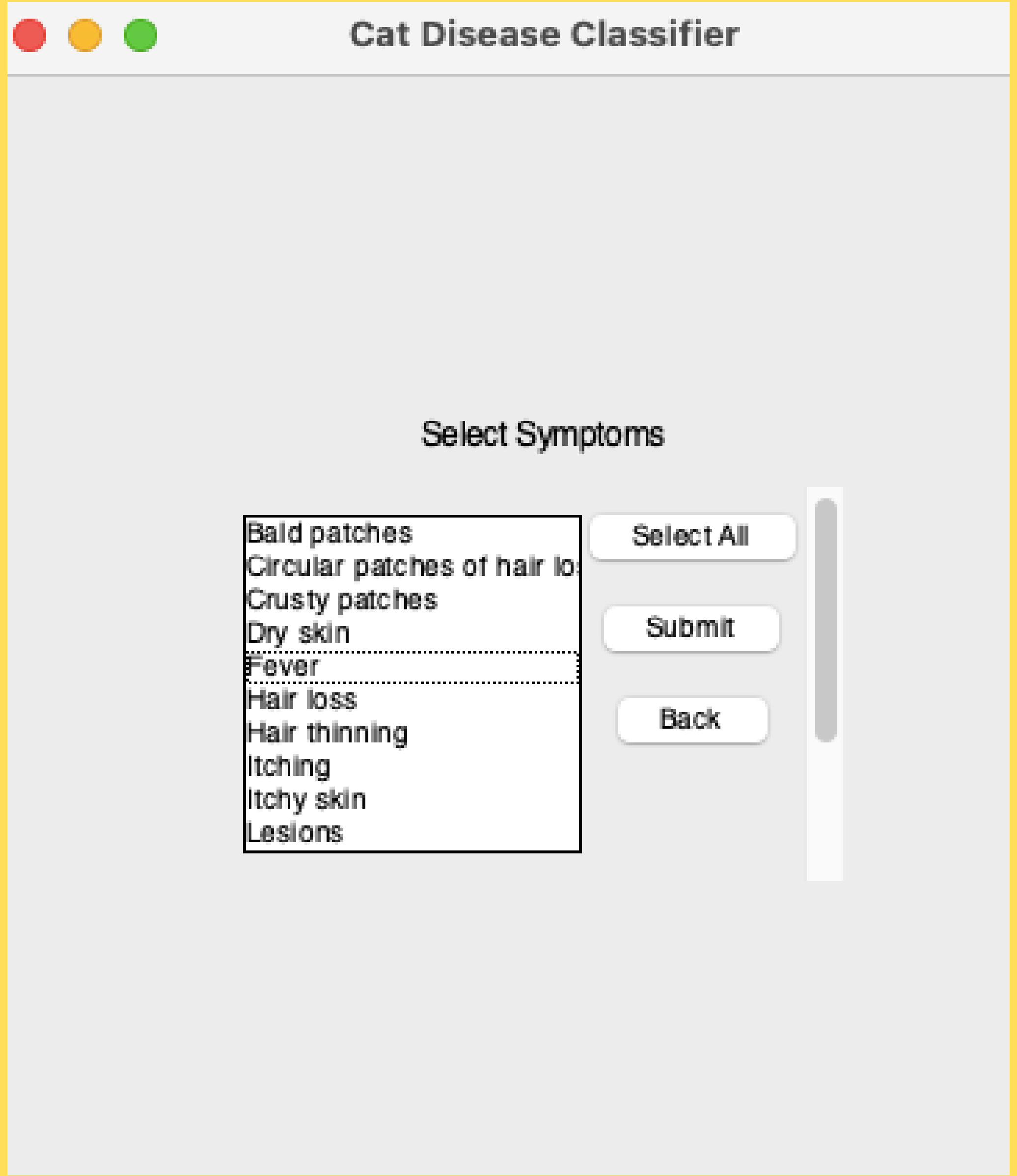
```
Image counts by class (including subfolders):

Class 'Birds': 307 images
Class 'Reptiles': 6044 images
Class 'Wild': 10477 images

Total number of images: 16828
Total number of classes: 3
```

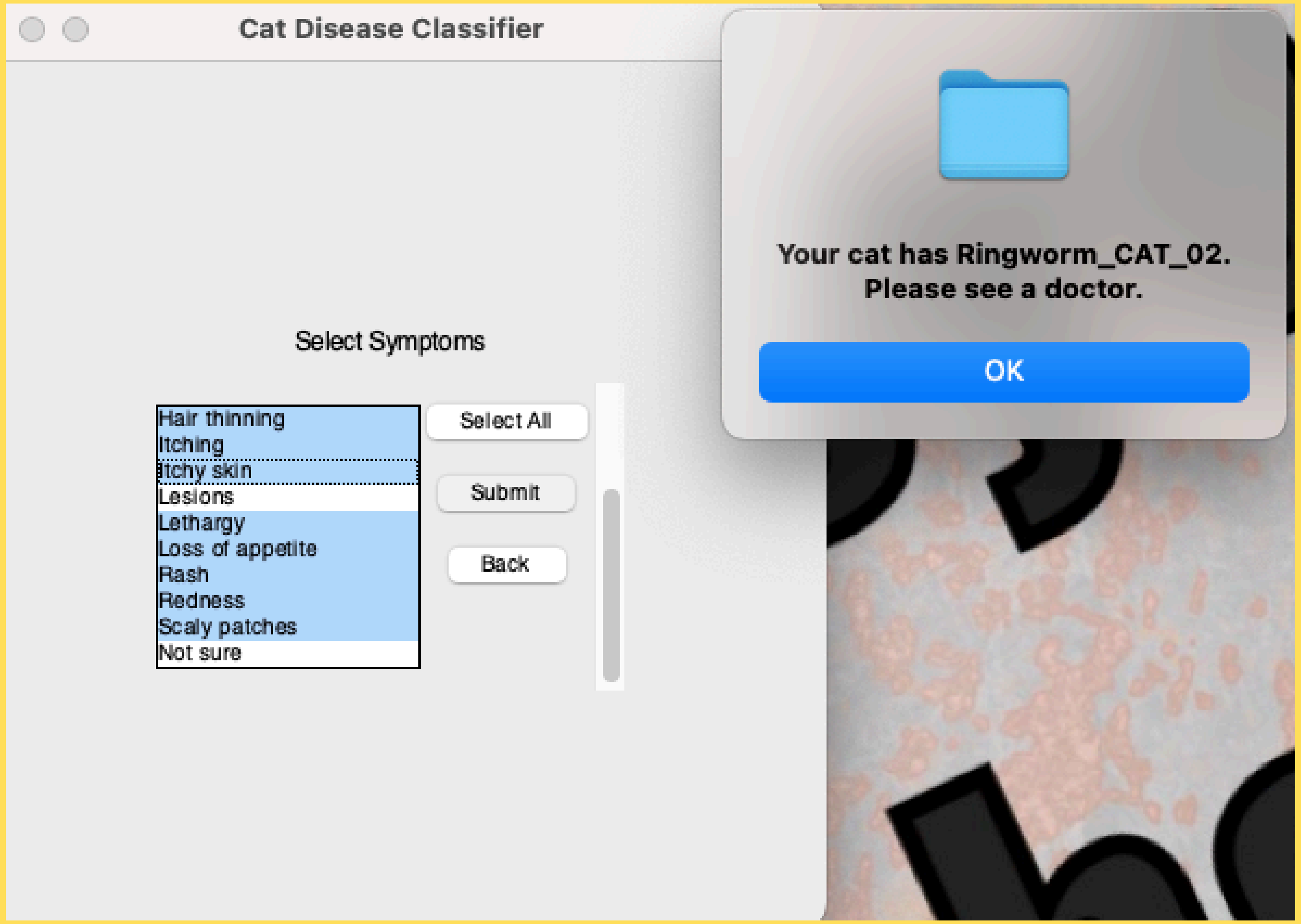




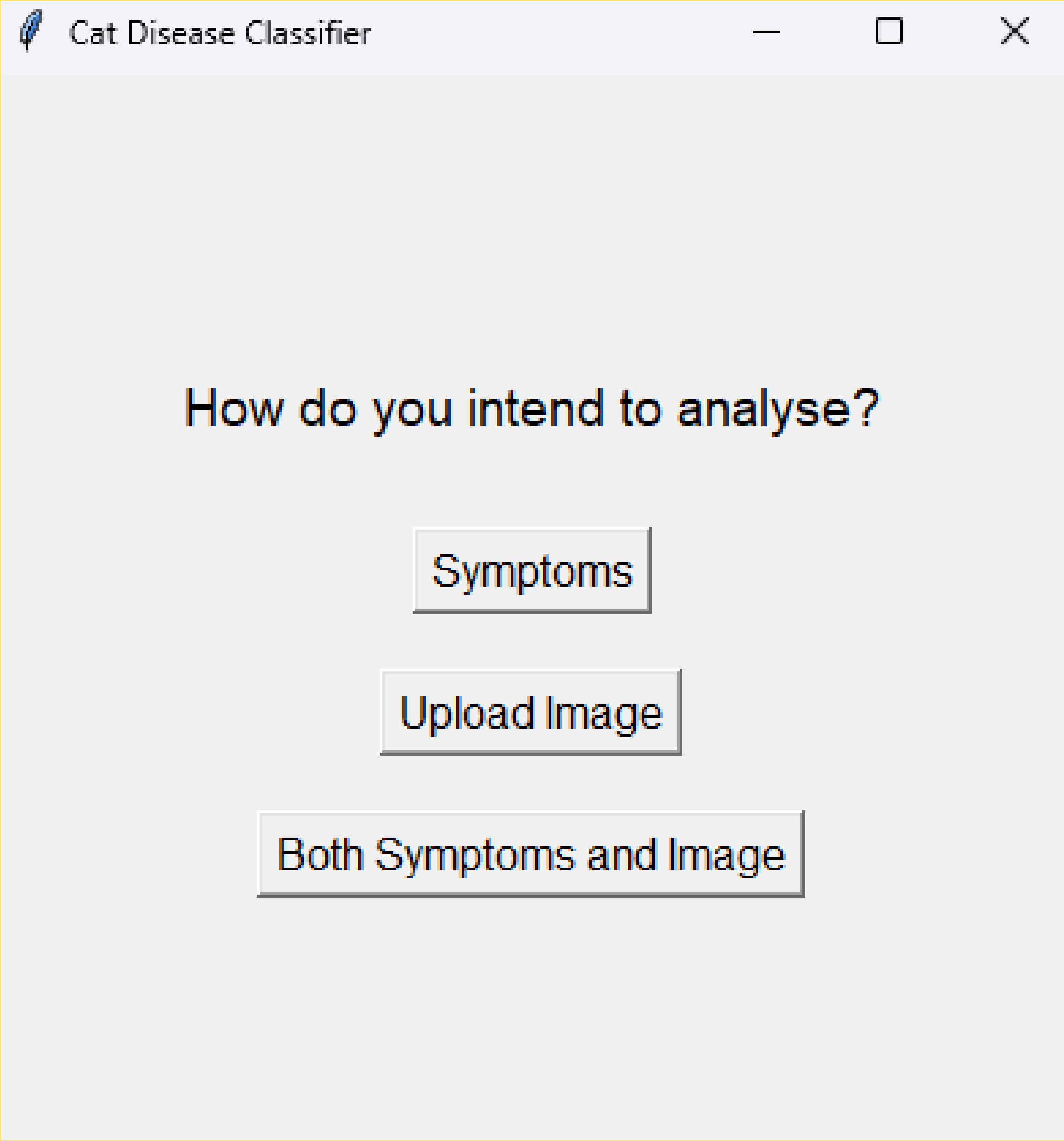
RESULTS

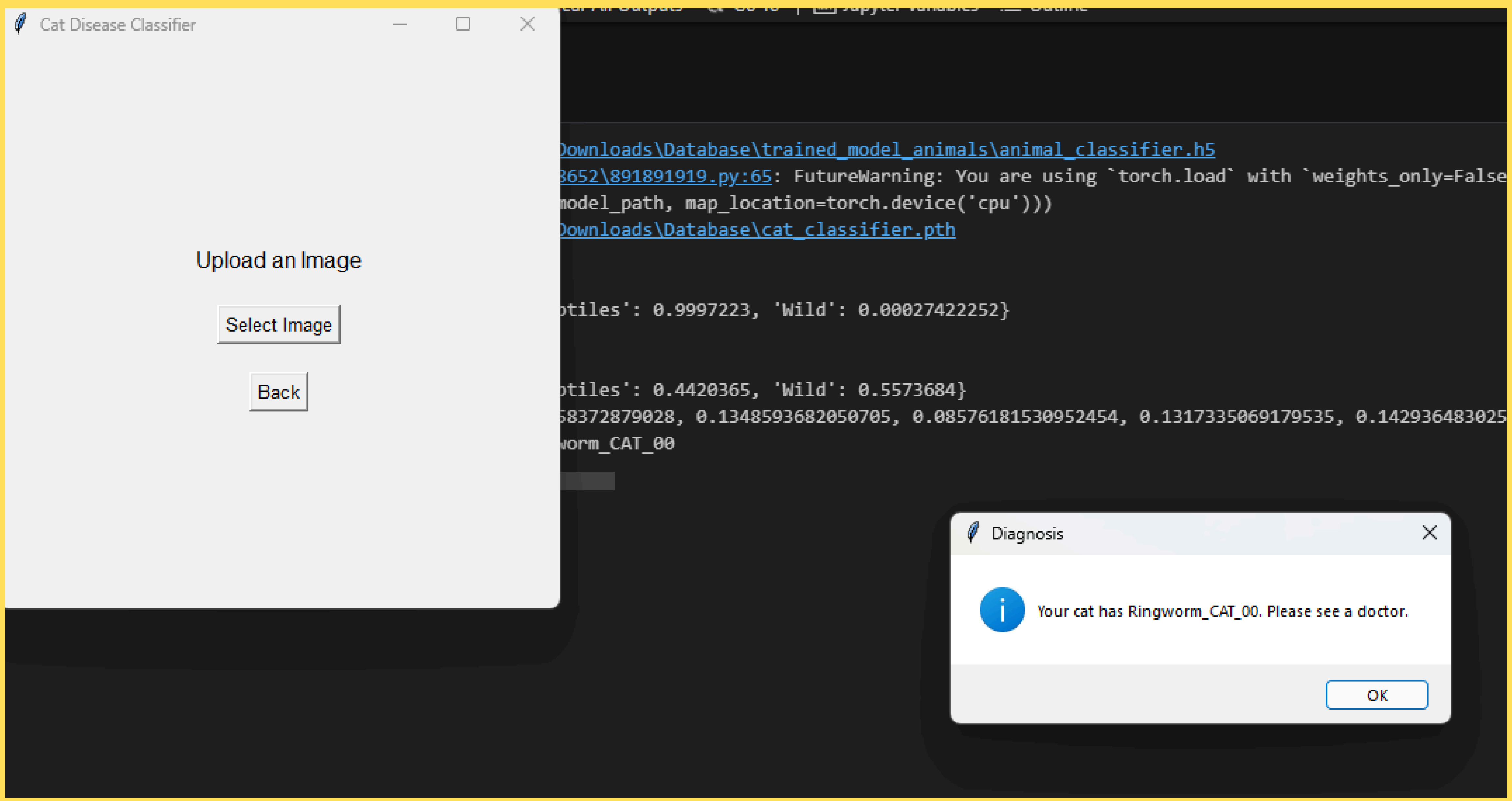

Cat Disease Classifier
How do you intend to analyse?
Symptoms
Upload Image
Both Symptoms and Image


Cat Disease Classifier
Select Symptoms
Bald patches
Circular patches of hair lo
Crusty patches
Dry skin
Fever
Hair loss
Hair thinning
Itching
Itchy skin
Lesions
Select All
Submit
Back

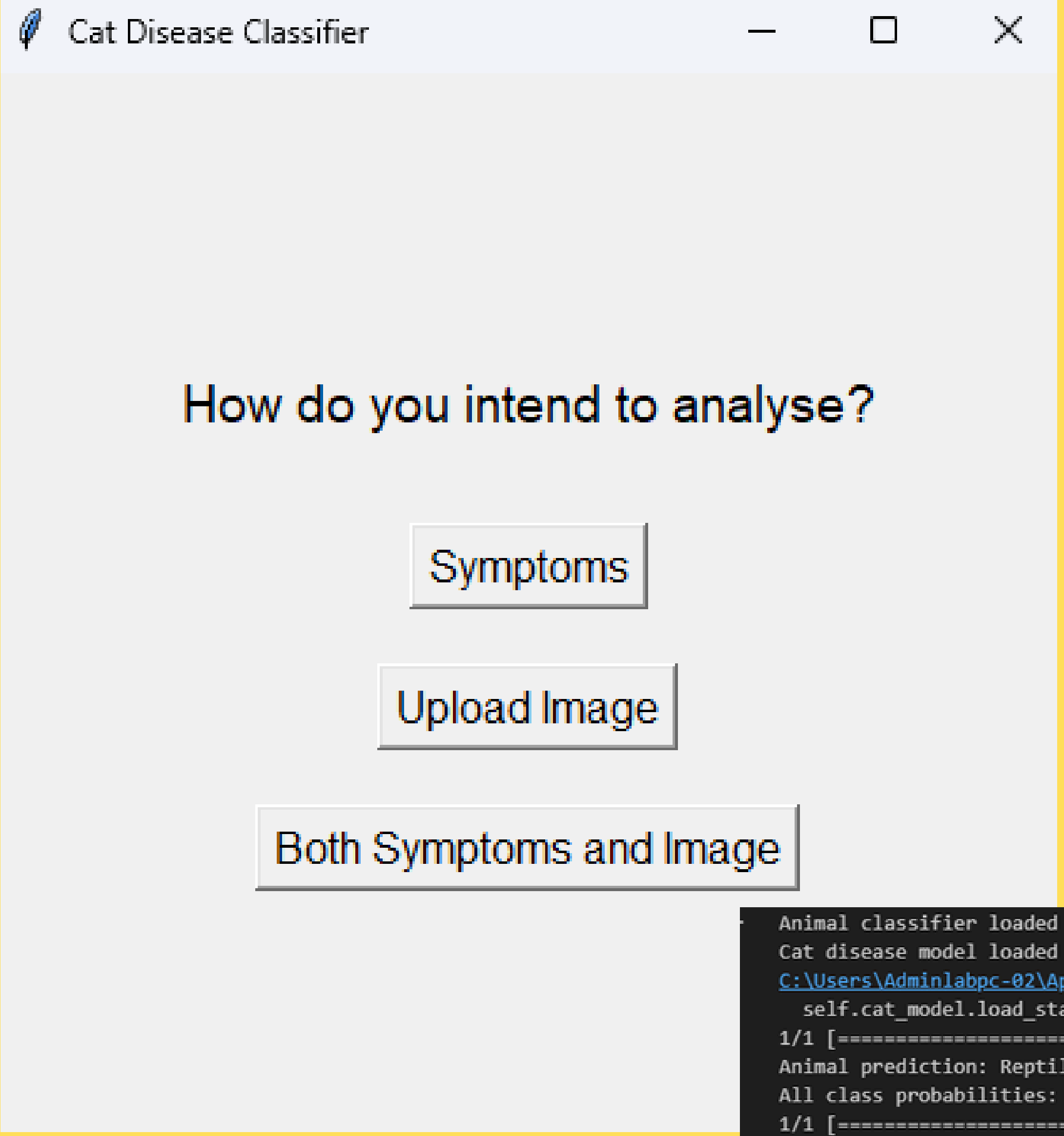

Cat Disease Classifier
Select Symptoms
Hair thinning
Itching
Itchy skin
Lesions
Lethargy
Loss of appetite
Rash
Redness
Scaly patches
Not sure
Select All
Submit
Back
Your cat has Ringworm_CAT_02.
Please see a doctor.
OK

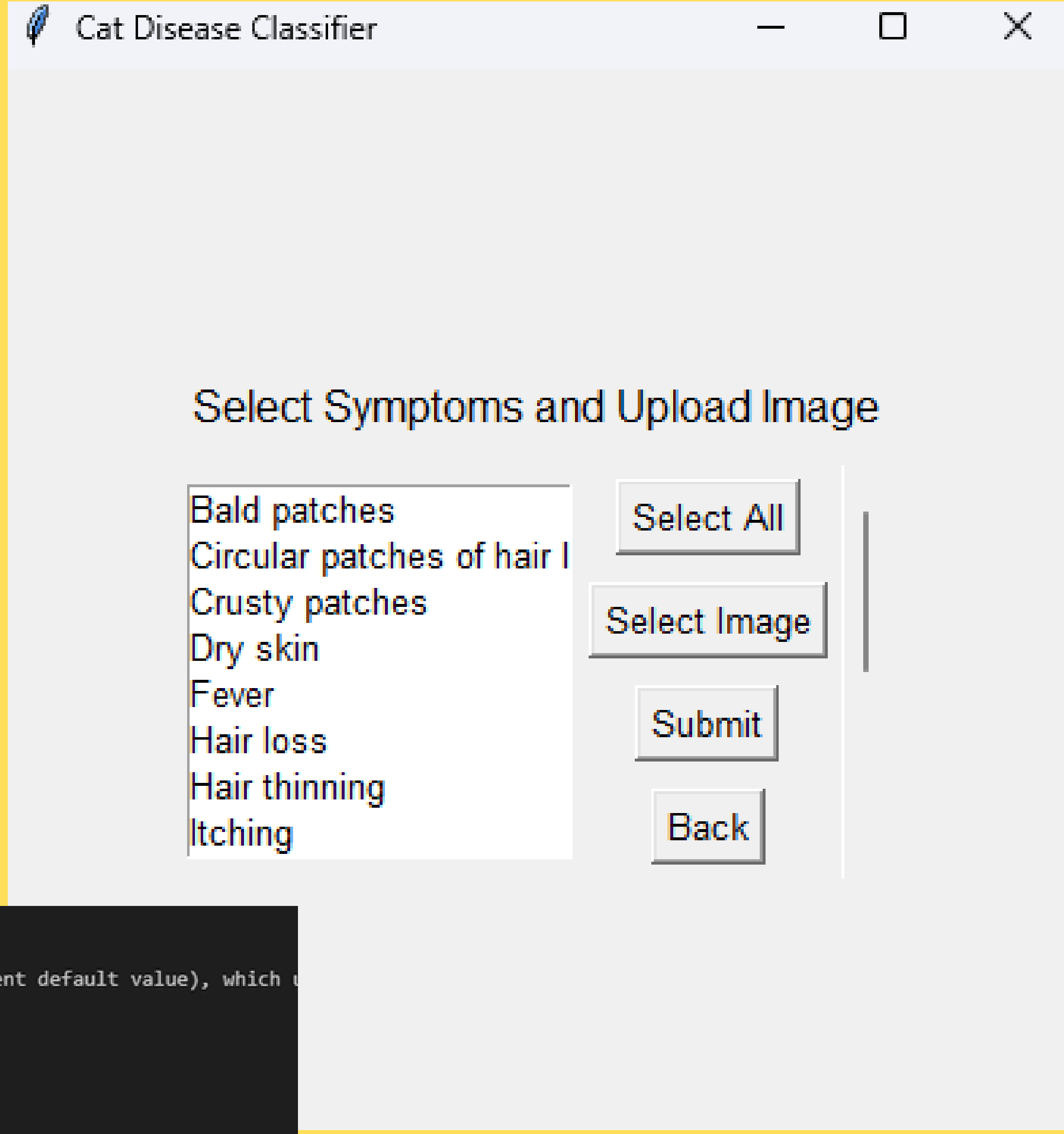
RESULTS

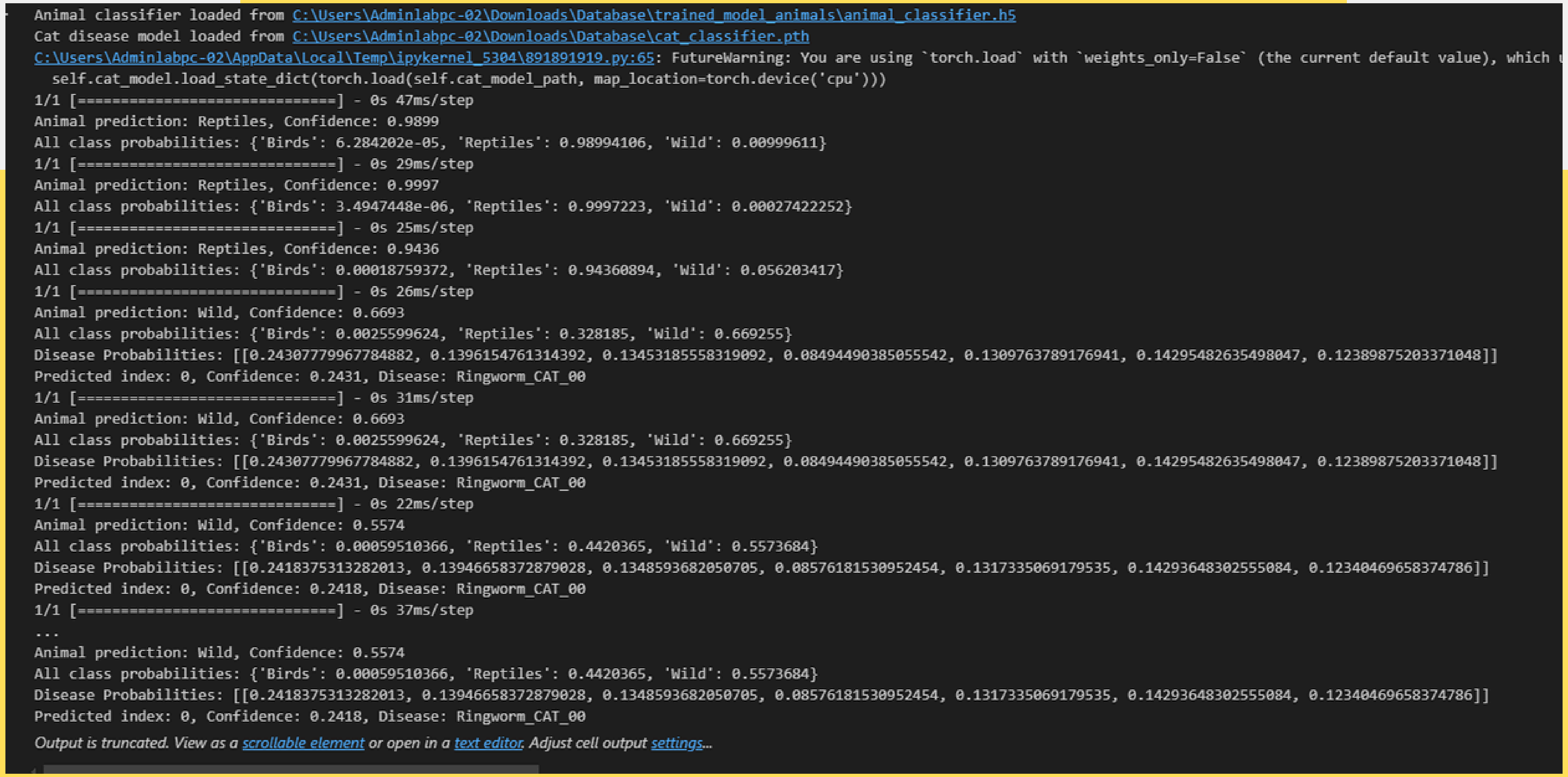

Cat Disease Classifier
How do you intend to analyse?
Symptoms
Upload Image
Both Symptoms and Image


Cat Disease Classifier
Upload an Image
Select Image
Back
Downloads\Database\trained_model_animals\animal_classifier.h5
3652\891891919.py:65: FutureWarning: You are using `torch.load` with `weights_only=False
model_path, map_location=torch.device('cpu')))
Downloads\Database\cat_classifier.pth
ptiles': 0.9997223, 'Wild': 0.00027422252}
ptiles': 0.4420365, 'Wild': 0.5573684}
58372879028, 0.1348593682050705, 0.08576181530952454, 0.1317335069179535, 0.142936483025
worm_CAT_00
Diagnosis
Your cat has Ringworm_CAT_00. Please see a doctor.
OK

RESULTS







```
Animal classifier loaded from C:\Users\Adminlabpc-02\Downloads\Database\trained_model_animals\animal_classifier.h5
Cat disease model loaded from C:\Users\Adminlabpc-02\Downloads\Database\cat_classifier.pth
C:\Users\Adminlabpc-02\AppData\Local\Temp\ipykernel_5304\891891919.py:65: FutureWarning: You are using `torch.load` with `weights_only=False` (the current default value), which
  self.cat_model.load_state_dict(torch.load(self.cat_model_path, map_location=torch.device('cpu')))
1/1 [==============================] - 0s 47ms/step
Animal prediction: Reptiles, Confidence: 0.9899
All class probabilities: {'Birds': 6.284202e-05, 'Reptiles': 0.98994106, 'Wild': 0.00999611}
1/1 [==============================] - 0s 29ms/step
Animal prediction: Reptiles, Confidence: 0.9997
All class probabilities: {'Birds': 3.4947448e-06, 'Reptiles': 0.9997223, 'Wild': 0.00027422252}
1/1 [==============================] - 0s 25ms/step
Animal prediction: Reptiles, Confidence: 0.9436
All class probabilities: {'Birds': 0.00018759372, 'Reptiles': 0.94360894, 'Wild': 0.056203417}
1/1 [==============================] - 0s 26ms/step
Animal prediction: Wild, Confidence: 0.6693
All class probabilities: {'Birds': 0.0025599624, 'Reptiles': 0.328185, 'Wild': 0.669255}
Disease Probabilities: [[0.24307779967784882, 0.1396154761314392, 0.13453185558319092, 0.08494490385055542, 0.1309763789176941, 0.14295482635498047, 0.12389875203371048]]
Predicted index: 0, Confidence: 0.2431, Disease: Ringworm_CAT_00
1/1 [==============================] - 0s 31ms/step
Animal prediction: Wild, Confidence: 0.6693
All class probabilities: {'Birds': 0.0025599624, 'Reptiles': 0.328185, 'Wild': 0.669255}
Disease Probabilities: [[0.24307779967784882, 0.1396154761314392, 0.13453185558319092, 0.08494490385055542, 0.1309763789176941, 0.14295482635498047, 0.12389875203371048]]
Predicted index: 0, Confidence: 0.2431, Disease: Ringworm_CAT_00
1/1 [==============================] - 0s 22ms/step
Animal prediction: Wild, Confidence: 0.5574
All class probabilities: {'Birds': 0.00059510366, 'Reptiles': 0.4420365, 'Wild': 0.5573684}
Disease Probabilities: [[0.2418375313282013, 0.13946658372879028, 0.1348593682050705, 0.08576181530952454, 0.1317335069179535, 0.14293648302555084, 0.12340469658374786]]
Predicted index: 0, Confidence: 0.2418, Disease: Ringworm_CAT_00
1/1 [==============================] - 0s 37ms/step
...
Animal prediction: Wild, Confidence: 0.5574
All class probabilities: {'Birds': 0.00059510366, 'Reptiles': 0.4420365, 'Wild': 0.5573684}
Disease Probabilities: [[0.2418375313282013, 0.13946658372879028, 0.1348593682050705, 0.08576181530952454, 0.1317335069179535, 0.14293648302555084, 0.12340469658374786]]
Predicted index: 0, Confidence: 0.2418, Disease: Ringworm_CAT_00
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
```

CONCLUSION & FUTURE WORK

RIMBERIO PET

Project Achievements

- Built a two-stage AI pipeline for early cat disease detection
- Achieved 98.7% accuracy for cat identification and 85% combined accuracy for disease classification
- Demonstrated the power of multi-modal learning (images + symptoms)
- Validated robustness through cross-validation and real-world photo testing

Potential Impact

- Offers a step toward affordable, accessible at-home vet screening
- Especially valuable for rural and underserved communities
- Encourages earlier vet visits and better health outcomes

Future Work

- Incorporate transformer-based models for richer symptom understanding
- Expand disease coverage and dataset diversity
- Develop a user-friendly app interface for non-expert use
- Integrate active learning to improve model over time with user feedback

THANK YOU!!