DEGREE: MSc in Artificial Intelligence

Module: Multi-Modal Chatbots

Assignment Title: Developing an Early Plant-Disease Detection Multi-Modal Chatbot- **PlantGuard**

Assignment Type: Set exercise

Word Limit: 2500-3000 words

Weighting: 100%

Issue Date: 4/9/2025

Submission Date: 3/10/2025

Feedback Date: 24/10/2025

Plagiarism:

When submitting work for assessment, students should be aware of the InterActive/Canvas guidance and regulations in concerning plagiarism. All submissions should be your own, original work.

You must submit an electronic copy of your work. Your submission will be electronically checked.

Learner declaration
I certify that the work submitted for this assignment is my own and research sources are fully acknowledged. **Student signature:** **Date:**

Harvard Referencing:

The Harvard Referencing System must be used. The Wikipedia, UKEssays.com or similar websites must **not** be used or referenced in your work.

Overview:

Timely and accurate identification of foliar diseases is essential for protecting crop yields and reducing unnecessary chemical treatments. However, small-scale farmers often rely on manual visual inspection, which can lead to **delayed, inaccurate, or inconsistent diagnoses**.
In this assignment, you will develop **PlantGuard** — a **proof-of-concept multimodal chatbot** designed to support early detection of plant diseases through **AI-powered analysis**. Your system must be capable of:

- Processing a **photograph of a symptomatic leaf** taken with a mobile device
- Capturing a **spoken description** of observed symptoms
- Accepting **text-based follow-up questions** from the user

Learning Outcomes:

LO1. Innovate and design multi-modal chatbots that effectively process and respond to text, voice, and visual inputs, incorporate natural language processing and generation techniques to enhance conversational capabilities.

LO2. Effectively communicate and present comprehensive ethical considerations and regulatory compliance strategies for multi-modal chatbot development, addressing issues related to data privacy, bias mitigation, and responsible AI practices.

LO3. Engage in practical development projects, create and deploy multi-modal chatbots across various industries, gain hands-on experience in user-centric design, data integration, and address real-world challenges faced in the field.

Assignment Goals:

Objective:

You are tasked with developing a **multimodal chatbot** that processes:

- **Leaf images**
- **Voice descriptions of symptoms**
- **Text-based questions**

Your chatbot application should be able to:

- Diagnose common plant diseases (e.g., powdery mildew, blight)
- Recommend appropriate treatments
- Demonstrate the full AI pipeline, from data processing and model development to deployment and ethical analysis

Assignment Tasks and Weighting:

1. Environment Setup (10%)

You are required to set up a suitable development environment that supports multimodal AI

applications.

- **Libraries:** PyTorch, TorchVision, Transformers (Hugging Face), OpenCV, SpeechRecognition, and either Streamlit or Flask. If opting for a browser-based interface, you may also utilize **JavaScript** along with **HTML** and **CSS**, supported by frameworks or libraries such as **React.js** or **Vue.js** for a dynamic frontend.
- **API Keys:** Configure access to cloud-based APIs if using services like Google Speech-to-Text or Hugging Face Inference API.

2. Data Acquisition & Exploration (15%)

2.1. Visual Data

- **Dataset:** PlantVillage (54 000+ leaf images, 38 classes) or any other publicly available leaf image datasets like ImageNet-1K datasets.
- **Sub-Task:**
  - Load and display sample images
  - Plot class distributions and compare representative healthy vs. diseased leaves.

2.2. Voice/Text Data

- **Synthetic Voice Descriptions:** Use synthetic audio recordings to simulate farmers describing plant symptoms. You may generate these using any free and publicly available online platforms, such as **LuvVoice**. Ensure that the recordings are clear, contextually relevant, and diverse enough to reflect different symptom descriptions.
- **FAQ Corpus:** Text Q&A pairs on plant-care best practices.
- **Sub-Task:**
  - Transcribe a selection of audio files and tokenize the text data
  - Explore vocabulary and semantic coverage across question categories

3. Preprocessing (15%)

3.1. Image Pipeline

- Resize and normalize the input data.
- Apply augmentation techniques such as horizontal flips, rotations, and color jitter to improve generalization

3.2. Audio Pipeline

- Convert WAV files to Mel-Frequency Cepstral Coefficients (**MFCCs)** features using librosa or a similar tool.

3.3. Text Pipeline

- Apply different NLP techniques to clean and structure the input text such as Lowercase, Stopword filtering, tokenization, etc.,

4. Model Design (20%)

4.1. Vision Model

- Fine-tune a pre-trained convolutional neural network (e.g., ResNet50/101 or any other suitable architecture such as EfficientNet, DenseNet, or MobileNet-V2 etc.) on any publicly available leaf image datasets like PlantVillage (via TensorFlow Datasets) or ImageNet-1K dataset to boost disease detection accuracy.
- Adapt the final classification layer to match the number of disease categories in your target dataset, if required.

- For models like ResNet-50 or ResNet-101, consider freezing the backbone layers during the initial training epochs to preserve learned representations while stabilizing the classification head.

4.2. Speech & Text Models

- **Speech**: Develop a CNN-LSTM classifier using **MFCCs** to classify voice-based symptom descriptions.
- **Text:** Fine-tune a BERT-based question answering model on a curated plant-care FAQ dataset to support natural language queries.

4.3. Multi-Modal Fusion

- Extract semantic embeddings from both the vision and text pipelines
- Concatenate the embeddings and feed them into a Multilayer Perceptron (MLP) with a unified classification head to generate the final disease prediction.

5. Training & Evaluation (20%)

5.1. Training Loops

- Design **independent training loops** for each modality (image, audio, text)
- Implement a final **joint fine-tuning phase** for the multimodal architecture
- Optionally use **TensorBoard** or similar tools to log and track training metrics.

5.2. Evaluation Metrics

- **Vision Model:** Accuracy, F1 per class, etc,
- **Speech/Text Models:**
  - WER (Word Error Rate) for speech → text,
  - Exact Match (EM) and F1-score for text QA.
- **Multi-Modal:** Overall diagnostic accuracy.

6. Deployment & User Testing (10%)

6.1 UI Prototype: You are expected to develop a professional, user-friendly web application that enables multimodal interaction with your chatbot. You may choose from one of the following technologies for frontend development:

- **Python-based frameworks:** such as **Streamlit** or **Flask**
- **Web technologies:** such as **JavaScript** in combination with **HTML** and **CSS**

Regardless of the chosen technology stack, your user interface should include the following core components:

- **Image Upload Widget:** to allow users to upload photographs (e.g., leaf images) for visual analysis.
- **Voice Input Widget:** to record or upload audio descriptions of symptoms.
- **Text Input Field:** to enable users to submit textual queries for follow-up questions or additional clarification.

The interface must ensure seamless, intuitive, and responsive user interactions across all modalities (image, voice, text). Accessibility and cross-browser compatibility are encouraged.

6.2 Containerization:

You must containerize your application using:

- A Dockerfile to define the runtime environment and install dependencies
- A docker-compose configuration to orchestrate multiple services (e.g., model backend, UI frontend).

This ensures **reproducibility**, **portability**, and alignment with **real-world deployment standards**.

7. Ethical & Regulatory Considerations (10%)

You are expected to address the following ethical and legal dimensions:

- **Data Privacy:** Ensure voice recordings comply with GDPR or equivalent regulations. No personally identifiable data should be stored.
- **Bias Mitigation:** Address dataset imbalance (e.g., underrepresented disease classes) using techniques such as **SMOTE** or **class reweighting**, if required.
- **Responsible AI:**
    - Provide confidence scores for each prediction
    - Clearly state that this tool provides agronomic advice, not medical or professional diagnoses

Submission Guidelines

Prepare a comprehensive report showcasing your chatbot design, implementation, and evaluation:

- Include screenshots or embedded visuals illustrating UI designs, model architecture, training progress, and evaluation metrics.
- Ensure all code is well-commented with clear replication instructions.
- Your report must be clear, organized, and visually appealing, using the **BSBI assignment template** available on Canvas.
- Upload your submission as a single file (PDF or DOC) on the BSBI portal.
- Python scripts or Jupyter notebooks should be uploaded to a repository platform (e.g., GitHub) with a **shared link included**.
- Cite all sources using the Harvard Referencing System.
- Submit your assignment electronically by the specified deadline.