GitaGPT Humanoid Robot Setup Instructions

This guide will help you set up and run the GitaGPT system for Bhagavad Gita Q&A with robotic jaw movement. It is written for beginners and covers all essential steps, file locations, and configuration details.

---

1. System Overview

- **Server (Windows Laptop):**
  - Main file: `server_v6.py`
  - Handles AI, speech-to-text, verse search, and text-to-speech
- **Client (Raspberry Pi):**
  - Main file: `client5.py`
  - Records audio, sends questions, plays answers, controls Arduino jaw

---

2. Required Files & Where to Place Them

On Windows (Server)

- **Example path shown below is for this guide only.**
- You must use your own Windows username and folder structure.
- To find your correct path:
  - Open File Explorer
  - Navigate to your Desktop or chosen folder
  - Copy the full path from the address bar
- Place these files in your chosen project folder, for example:
  `C:\Users\<YOUR_USERNAME>\OneDrive\Desktop\gitaGPT\`
  - `server_v6.py` (main server code)
  - `bhagavad_gita_verses.csv` (Gita verses data)
  - `gita_faiss.index` (auto-generated FAISS index)
  - `en_GB-southern_english_female-low.onnx` (Piper TTS voice model)

On Raspberry Pi (Client)

- Place these files in your chosen Pi project folder, for example:
  `/home/pi/test/`
  - `client5.py` (main client code)

Tip:

- You can use any folder you like, but you must update the file paths in your code to match where you put the files.
- If you are unsure, ask for help or send a screenshot of your folder location.

---

3. How to Change File Paths in server_v6.py

If your files are in different locations, you must update the following paths in `server_v6.py`:

- **FAISS Index Path:**
  - Find the line:

```
FAISS_INDEX_PATH = os.environ.get("FAISS_INDEX_PATH",
r"C:\Users\Raghuram S\OneDrive\Desktop\gitaGPT\gita_faiss.index")
```

  - Change the path to where your `gita_faiss.index` file is located.
- **CSV Path:**
  - Find the line:

```
DF_PATH = os.environ.get("DF_PATH", r"C:\Users\Raghuram
S\OneDrive\Desktop\gitaGPT\bhagavad_gita_verses.csv")
```

  - Change the path to where your `bhagavad_gita_verses.csv` file is located.
- **Piper TTS Voice Path:**
  - Find the line:

```
PIPER_VOICE_PATH = os.environ.get("PIPER_VOICE_PATH",
r"C:\Users\Raghuram S\OneDrive\Desktop\gitaGPT\en_GB-
southern_english_female-low.onnx")
```

  - Change the path to where your Piper `.onnx` voice file is located.

Tip:

- You can use absolute paths (full path from C:\ or /home/pi/) or relative paths (from your project folder).
- Always use double backslashes `\\` in Windows paths, or single slashes `/` in Linux paths.

---

4. IP Address and Port Configuration (Client)

- **Find your laptop's IP address:**
  - On Windows, open Command Prompt and run: `ipconfig`
  - Look for your WiFi IPv4 address (e.g., `192.168.104.80`)

4. Update the Client Code: IP Address, Port, and Arduino Configuration

Before running the client, open your client Python file (e.g., `humanoid_gita_client.py` or `Client5.py`) and update the configuration section at the top to match your setup:

```
# Configuration
LAPTOP_IP = "192.168.104.80"  # Your laptop IP
LAPTOP_PORT = 5000
SAMPLE_RATE = 16000
CHANNELS = 1
RECORD_SECONDS = 10

# Arduino configuration (from your original code)
ARDUINO_PORT = "/dev/ttyUSB0"  # Change as needed
ARDUINO_BAUDRATE = 9600
```

- **LAPTOP_IP**: Set to your Windows laptop's IP address (find with `ipconfig` on Windows).
- **LAPTOP_PORT**: Should match the port number used by your server (default: 5000).
- **ARDUINO_PORT**: Set to the correct serial port for your Arduino (use `ls /dev/tty*` on Raspberry Pi to find it).
- **ARDUINO_BAUDRATE**: Usually 9600, but match your Arduino sketch if different.

If you get errors like "Connection refused" or "Serial port not found," double-check these settings and restart the client.

4a. Ollama Installation & Setup (Server)

Ollama is used for advanced AI text generation in `server_v6.py`. You must install and set up Ollama on your Windows laptop before starting the server.

Steps to Download and Install Ollama:

1. **Go to the Ollama website:**
   - https://ollama.com/download
2. **Download the Windows installer** and run it.
3. **Follow the installation instructions** to complete setup.

Steps to Set Up Ollama for GitaGPT:

1. **Open PowerShell and start the Ollama server:**

   ```
   ollama serve
   ```

   - This command must be running in the background while you use GitaGPT.
2. **Download the required model (e.g., Gemma):**

   ```
   
   ```

```
ollama pull gemma3:1b
```

   - You can pull other models if needed. See https://ollama.com/library for options.
3. **(Optional) Allow remote access:**
   - If you want to access Ollama from another device, set the environment variable before starting Ollama:

     ```
     $env:OLLAMA_HOST="0.0.0.0:11434"
     ollama serve
     ```

   - This allows connections from other devices on your network.
4. **Check Ollama status:**
   - Visit http://localhost:11434 in your browser to confirm Ollama is running.

Note:

- If you do not install or run Ollama, the server will use a fallback model (transformers), which may be slower or less accurate.
- For more help, see https://ollama.com/docs/getting-started

---

4. Virtual Environment Setup & Python Package Installation

On Windows (Server)

1. Open PowerShell and navigate to your project folder:

   ```
   cd "C:\Users\<YOUR_USERNAME>\OneDrive\Desktop\gitaGPT"
   ```

2. Create a virtual environment:

   ```
   python -m venv gitagpt
   ```

3. Activate the virtual environment:

   ```
   .\gitagpt\Scripts\Activate.ps1
   ```

4. Upgrade pip (recommended):

   ```
   python.exe -m pip install --upgrade pip
   ```

5. Install all required Python packages:

```
pip install flask numpy openai-whisper sentence-transformers faiss-cpu
pandas scipy pyttsx3 piper-tts soundfile transformers comtypes pywin32
pyserial ollama
```

   - Only `pyserial` is needed for serial communication; you do NOT need to install `serial` or `serial.tools` separately.

On Raspberry Pi (Client)

1. Update and install Python3/pip:

```
sudo apt update
sudo apt install python3 python3-pip python3-venv
```

2. Create a virtual environment:

```
python3 -m venv gitaenv
```

3. Activate the virtual environment:

```
source gitaenv/bin/activate
```

4. Upgrade pip (recommended):

```
python3 -m pip install --upgrade pip
```

5. Install all required Python packages:

```
pip install sounddevice pyaudio pyserial requests
```

Note:

- You only need to install these packages once per environment.
- If you see errors about missing modules, run the relevant pip install command above.

---

5. Customizing IP Address, Port, and Arduino Serial Port

After installing packages and before running the client, you must update the following settings in your client code (e.g., `humanoid_gita_client.py` or `Client5.py`) on your Raspberry Pi:

How to Edit

1. Open your client Python file in a text editor (e.g., VS Code, nano, Thonny).
2. Find the section near the top that looks like this:

```
# Configuration
LAPTOP_IP = "192.168.104.80"  # Your laptop IP
LAPTOP_PORT = 5000
SAMPLE_RATE = 16000
CHANNELS = 1
RECORD_SECONDS = 10

# Arduino configuration (from your original code)
ARDUINO_PORT = "/dev/ttyUSB0"  # Change as needed
ARDUINO_BAUDRATE = 9600
```

What to Change

- **LAPTOP_IP**: Set this to your Windows laptop's IP address on the local network. Find it using `ipconfig` (Windows) or `ifconfig` (Linux/Raspberry Pi).
- **LAPTOP_PORT**: Should match the port number used by your server (default: 5000).
- **ARDUINO_PORT**: Set this to the correct serial port for your Arduino. Common values:
  - `/dev/ttyUSB0` (most USB Arduinos)
  - `/dev/ttyACM0` (some models)
  - Use `ls /dev/tty*` on Raspberry Pi to find the correct port.
- **ARDUINO_BAUDRATE**: Usually 9600, but match your Arduino sketch if different.

When and Why to Change

- Change these values **before running the client** for the first time, or whenever your network or hardware setup changes.
- If you get errors like "Connection refused" or "Serial port not found," double-check these settings.

Troubleshooting

- If the client cannot connect to the server, verify LAPTOP_IP and LAPTOP_PORT match your server's actual IP and port.
- If you see errors about the Arduino port, run `ls /dev/tty*` on the Raspberry Pi to find the correct device name, then update ARDUINO_PORT.
- Restart the client after making changes.

---

6. Running the System

On Server (Windows Laptop)

1. Activate your virtual environment:

```
.\gitagpt\Scripts\Activate.ps1
```

2. Start the server:

```
python server_v6.py
```

3. Wait for the "Server ready" message and ensure no errors.

On Client (Raspberry Pi)

1. Activate your virtual environment:

```
source gitaenv/bin/activate
```

2. Run the client script:

```
python3 client5.py
```

3. The client will automatically:
    - Connect to the Arduino (if plugged in)
    - Handle audio recording and playback
    - Control the jaw movement
    - Communicate with the server
4. Follow on-screen instructions:
    - Press ENTER to record a question
    - Speak clearly for 10 seconds
    - Watch the jaw move during Gita's response

Note:

- You do NOT need to manually connect to the Arduino or test TTS before running the client script. Everything is handled by the script.

---

7. Important Things to Change (Summary)

- **IP Address:**
    - Change the IP in the client code (`LAPTOP_IP` variable) to match your laptop's IP.

- **Server Port:**
  - Change the port in the client code (`SERVER_PORT` variable) if your server uses a different port.
- **File Paths:**
  - Make sure all files are placed in the correct folders as shown above.
  - Update FAISS index, CSV, and Piper voice paths in `server_v6.py` if your files are in different locations (see section 9).
- **Virtual Environment:**
  - Always activate the venv before running any Python scripts.

---

8. Troubleshooting

- **Arduino Not Responding:**
  - Check USB connection and power
  - Try both `/dev/ttyUSB0` and `/dev/ttyUSB1` in your client code
  - Re-upload Arduino code if needed
- **Audio Issues:**
  - Ensure microphone is connected to Pi
  - Test with `arecord` and `aplay` commands
- **Virtual Environment Problems:**
  - Always activate the venv before running scripts
- **General:**
  - If you see errors, check file paths, IP addresses, and package installations

---

9. Summary

- Use **only** `server_v6.py` (server) and `client5.py` (client) for the final system.
- Update IP addresses and file paths for your own setup.
- Share this guide with anyone setting up the system—no prior experience required!
