Hey_Nova: Virtual Voice Assistant

Manasa Indu Sri

April 2025 – May 2025

Project Repository: `https://github.com/manasa123456-78/Hey_Nova`

Abstract

Hey_Nova is a Python-based virtual voice assistant capable of processing natural language queries, learning dynamic Q&A responses, automating system-level tasks, and interacting with Windows through speech commands. The assistant integrates multiple Python libraries to provide a modular, hands-free desktop experience.

Contents

1 Features

- Supports over 20 unique voice commands for desktop automation and hands-free workflows.
- Real-time natural language processing using SpeechRecognition.
- Dynamic Q&A learning with JSON-based persistent storage.
- Automates system tasks like browser tab control, fullscreen toggling, desktop display, and media control.
- Google search and YouTube playback via voice commands.
- Windows Action Center automation: toggles Bluetooth, Hotspot, and Night Light.
- Time-based greeting system with interactive, personalized dialogues.

2 File Structure

`main.py`	Core assistant loop and command handler.
`speech.py`	Speech recognition and text-to-speech engine.
`functions.py`	Web search, Q&A training, system controls.
`automation.py`	Windows Action Center automation.
`data.json`	Dynamic Q&A storage.

3 Evolution of Features

Step	Feature Added
1	Basic voice input and text-to-speech response
2	Integration of system control commands (show desktop, fullscreen)
3	Dynamic Q&A learning via JSON
4	Web search and YouTube playback automation
5	Windows Action Center control using pywinauto
6	Personalized time-based greeting logic

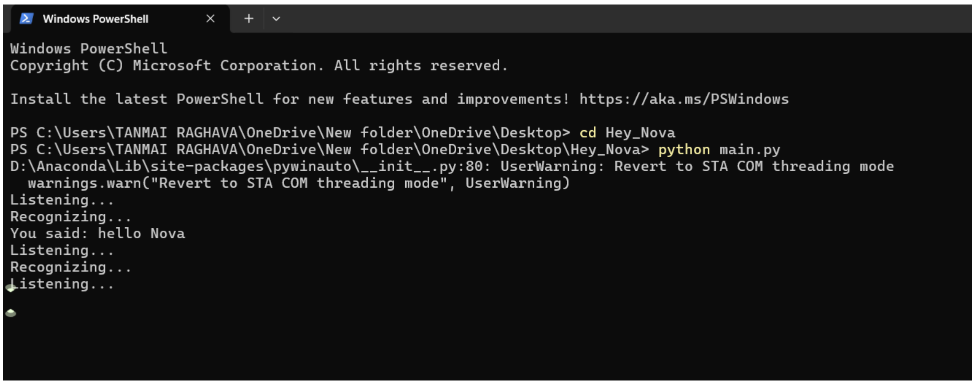
4 Commands Supported

- **Activation:** "Hello Nova" or "Hey Nova"
- **System:** "Show desktop", "Close tab", "Full screen", "Pause", "Resume"
- **Web:** "Search *X* on Google", "Play *Y* on YouTube"
- **Dynamic Learning:** "When I say *X* you say *Y*"
- **Action Center:** "Turn on Bluetooth", "Enable Hotspot", "Night Light"
- **Shutdown:** "Take rest", "Go home"

5 How It Works

- Nova continuously listens for wake words to start the assistant loop.
- Once activated, it greets the user based on system time (morning, afternoon, evening).
- Nova processes voice commands using Google's SpeechRecognition engine.

- Commands are parsed and mapped to system functions, web tasks, or knowledge base lookups.
- New Q&A entries can be added dynamically using voice commands.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\TANMAI RAGHAVA\OneDrive\New folder\OneDrive\Desktop> cd Hey_Nova
PS C:\Users\TANMAI RAGHAVA\OneDrive\New folder\OneDrive\Desktop\Hey_Nova> python main.py
D:\Anaconda\Lib\site-packages\pywinauto\__init__.py:80: UserWarning: Revert to STA COM threading mode
  warnings.warn("Revert to STA COM threading mode", UserWarning)
Listening...
Recognizing...
You said: hello Nova
Listening...
Recognizing...
Listening...
```

Figure 1: Command line interface during active session

6 Requirements

- Python 3.8+
- Libraries: SpeechRecognition, Pyttsx3, PyAutoGUI, Pywinauto, JSON (standard)

7 How to Run

1. Clone the repository: `git clone https://github.com/manasa123456-78/Hey_Nova`
2. Install dependencies: `pip install -r requirements.txt`
3. Run: `python src/main.py`

8 Ideas for Future Additions

- Multi-language voice recognition support.
- Integration with calendar and email APIs.
- GUI interface for real-time command visualization.
- Improved NLP-based query understanding.