
LanguageLeapAI



LEAP across Language barriers by using AI to converse with other online users from across the globe! **LanguageLeapAI** aims to provide you a real-time language AI assistant that can understand and speak your desired language fluently. *(Targeted towards English to Japanese and German as of right now)*

Setup Guide: <https://www.youtube.com/watch?v=bN5UaEkIPGM>

Showcase: <https://www.youtube.com/watch?v=UY7sRB60wZ4>

Integration of AI Entities

This project integrates 3 free and open-source AI systems:

1. WhisperAI: General-purpose Speech Recognition Model developed by OpenAI that can perform multilingual speech recognition.
2. DeepL Translator: Powered by neural networks and the latest AI innovations for natural-sounding translations
3. Voicevox: Japanese Deep-Learning AI Voice Synthesizer

WhisperAI and Voicevox both have docker images available on DockerHub, so we will be building and running them both via a Docker Compose file. DeepL can be interacted with by signing up for a free

plan and interacting with its REST API up to 500,000 character limit / month. If DeepL is unavailable in your country, an option to use Google Translate instead is available.



How it works

LanguageLeapAI is made up of 2 main python programs.

Voice Translator

The first, `voice_translator.py`, records your microphone whenever a push-to-talk key is held down on the keyboard. Once this key is released, it saves your voice in an audio file which is then sent to WhisperAI's transcribe endpoint which runs Automatic Speech Recognition (ASR) on it. After a response containing your speech as text is received, this text is then translated using DeepL's REST API.

The translated text is then sent to Voicevox which performs text-to-speech and generates an audio file voiced in Japanese. This file is then played to your target application's microphone input and your speakers/headphones.

Since Voicevox only takes in Japanese text as input and generates speech in Japanese, the project is technically only limited to Japanese as the target language. However, Voicevox can be replaced with any other text to speech program that can speak your desired language for limitless possibilities.

Thorsten has been added as a German TTS program.

Audio Subtitler

The second, `subtitler.py`, records your application's audio output and listens in the background for any speech. Once it has detected that a phrase/sentence is complete, it saves the audio into a wav file and sends it to WhisperAI's translate endpoint which translates the speech from the target language to English.

This English text is then displayed on screen using python's tkinter module, essentially acting as subtitles.

Applications

LanguageLeapAI's target audience is for users who want to chat with another but do not speak the same language. An example is an English-speaking user playing an online game in the Japan server but wants to use voice chat despite not knowing Japanese.

By running both `subtitler.py` and `voice_translator.py`, they can understand their fellow Japanese teammates by reading the english subtitles generated in real time. They can also speak English and the Japanese teammates will instead hear the translated Japanese speech generated by Voicevox.

However, this is not the only application of **LanguageLeapAI**.

Only using Audio Subtitler

User simply wants to understand what is being said with no need to speak. E.g. Watching a video / stream / movie in another language without subtitles. The user can choose to not run `voice_translator.py` and simply use `subtitler.py`.

Only using Voice Translator

User understands the language enough to listen and understand, but is afraid to speak the language for various reasons, e.g. Anonymity / Fear of messing up or offending. The user can choose to not run `subtitler.py` and simply use `voice_translator.py`.

Setup

Setting up **LanguageLeapAI** requires 3 crucial steps, so don't miss out on any of them! 1. Installing Services and Dependencies 2. Audio Routing 3. Writing your Environment file

Usage

To run **LanguageLeapAI**, you need to first run WhisperAI and Voicevox. They can either be run via Docker or using Google Colab.

Google Colab

If your GPU is not powerful enough, you may want to consider running WhisperAI and Voicevox using Google Colab's GPU.

Upload `run_whisper_n_voicevox.ipynb` file to Google drive, open the notebook with Google Colab and simply follow the instructions!

To run only whisper or voicevox on the cloud: Use either the `run_whisper_colab.ipynb` and `run_voicevox_colab.ipynb` Colab files instead!

Docker

Voicevox - JA If you still want to run both Whisper and Voicevox on your computer, run these commands in the folder containing the `docker-compose.yml` file.

To run both WhisperAI and Voicevox:

```
docker-compose up -d
```

To stop running the containers:

```
docker-compose down
```

If you're running Windows Subsystem for Linux (WSL) don't forget to shut it down to reclaim your ram. This should only after you have stopped the containers and are done using the program.

```
wsl --shutdown
```

TTS Thorsten - DE If you want to run a German version of Voicevox, you need to change the docker-compose file to the corresponding one. The TTS is the only thing that's changing, so make sure to also change the `TARGET_LANGUAGE_CODE` in your `.env` file.

To run both WhisperAI and Thorsten:

```
docker-compose -f docker-compose-de.yml up -d
```

To stop running the containers:

```
docker-compose down
```

Python Program

Run these commands in the src/ folder.

To run the Audio Subtitler:

```
python subtitler.py
```

To run the Voice Translator:

```
python voice_translator.py
```

To stop the python scripts, simply press `Ctrl+C` in the terminal.

Things to note

Some important things to keep in mind while using **LanguageLeapAI**.

Whisper's inconsistency Do note that WhisperAI is not exactly the most accurate and will not transcribe speech correctly 100% of the time, so use at your own risk. Until OpenAI decides to improve the dataset that was used to train the Whisper models, this will have to do.

Also, Whisper is not designed to handle multiple concurrent requests at once. However, for subtitles to be updated in time, multiple requests are being sent asynchronously, so some requests might return an error.

Antivirus Web Protection If you are running Whisper and Voicevox on the cloud using Google Colab, since we are using ngrok and localtunnel to host our services, the randomised public IP address that they provide might be blacklisted by your antivirus software. If the AI seems to stop working, it may be due to your antivirus blocking the connections to these public IP addresses. You may whitelist these IP addresses or just turn off your antivirus web protection **at your own risk**.

Voicevox voices There are certain terms and conditions for using the voices from Voicevox, so do read up on these before using a specific speaker.

License

The code of LanguageLeapAI is released under the MIT License. See LICENSE for further details.