

# Things to Look Into

- <https://picovoice.ai/blog/whisper-cpp-speaker-diarization/>

“[OpenAI Whisper](#) delivers highly accurate speech-to-text transcription, but it does not track speaker changes. Applications that rely on Whisper cannot determine *who* is speaking in a conversation. To add [speaker labels](#) such as "Speaker 1" and "Speaker 2," you can integrate [Falcon Speaker Diarization](#) with [whisper.cpp](#). [creating a fully local, offline, multi-speaker transcription system](#).

This tutorial explains how to add speaker segmentation to Whisper, so you can determine "who spoke when" and generate timestamps to produce labeled Whisper transcripts suitable for multi-speaker recordings. This approach is ideal for use cases such as [podcast transcriptions](#), [meeting transcription and summarization](#), or [call center analytics](#), where speaker identification is essential for readability and downstream analysis.

- <https://github.com/av/harbor>

Doesn't work with whisper.cpp? Would have been an interesting option.

“ Harbor is a CLI and companion app that lets you spin up a complete local LLM stack—backends like Ollama, llama.cpp, or vLLM, frontends like Open WebUI, plus supporting services like SearXNG for web search, Speaches for voice chat, and ComfyUI for image generation—all pre-wired to work together with a single `harbor up` command. No manual setup: just pick the services you want and Harbor handles the Docker Compose orchestration, configuration, and cross-service connectivity so you can focus on actually using your models.

- LocalAI <https://localai.io/>

This will definitely be a starting point on better hardware.

- whisper-diarization might make discerning the speaker possible
- 

Possible better(?) model for German? (Definitely not feasible on a Raspberry Pi 500)

<https://huggingface.co/primeline/whisper-large-v3-german>

---

Created 2026-01-11 12:28:43 UTC by Mela

Updated 2026-02-22 05:30:34 UTC by Mela