notebooklm-nonsense.mp4

Sally

[00:00:02.06] Sally: Okay, so today we're diving into something pretty critical

[00:00:04.97] Sally: in the modern tech landscape.

[00:00:08.10] Sally: How do you get really good AI transcription,

[00:00:10.65] Sally: fast, accurate, but crucially,

[00:00:13.09] Sally: without sending potentially sensitive stuff,

[00:00:15.13] Sally: audio video up to some third party cloud service?

[00:00:18.21] Sally: This is our deep dive into the documentation

[00:00:20.61] Sally: for a tool set called Transcribe with Whisper.

[00:00:23.29] Sally: It's all about self-hosting.

Joe

[00:00:25.04] Joe: Exactly. It's aiming to solve that core data sovereignty problem, keeping your data completely

[00:00:31.12] Joe: under your control. And what's really interesting here, the source material, the README,

Sally

[00:00:29.76] Sally: Hmm.

Joe

[00:00:35.55] Joe: States their goal pretty boldly. They want to make this advanced AI accessible,

[00:00:39.95] Joe: even if your technical skill is basically just installing programs and clicking buttons.

[00:00:44.51] Joe: That's their quote more or less.

Sally

[00:00:45.98] Sally: Hmm, that is ambitious, especially with AI tools they often have these really complex

[00:00:50.10] Sally: dependencies, right?

[00:00:51.10] Sally: So does the documentation actually show they pulled it off, hiding the complexity while,

[00:00:55.78] Sally: you know, keeping that privacy promise for the data?

Joe

[00:00:57.96] Joe: It seems like they really have.

[00:00:59.48] Joe: The unique part is how it all fits together.

[00:01:01.52] Joe: It's not just using a powerful model,

[00:01:03.44] Joe: like say, faster whisper for the accuracy.

[00:01:06.44] Joe: It intelligently bundles that with other vital things,

[00:01:10.72] Joe: like speaker separation, knowing who spoke when

[00:01:13.44] Joe: and making sure the output plays nice

[00:01:15.32] Joe: with research software or analytical tools,

[00:01:19.60] Joe: think beyond just basic text.

Sally

[00:01:21.30] Sally: Right, okay, let's translate that.

[00:01:23.54] Sally: So the process, it sounds clean enough,

[00:01:25.98] Sally: you feed it a video file MP4 MOV MKV,

[00:01:28.70] Sally: it turns that into an audio only Wav file first.

[00:01:32.06] Sally: Is that conversion step okay?

[00:01:33.58] Sally: Like no quality loss?

Joe

[00:01:34.98] Joe: Yeah, it's designed to be lossless, just stripping the video out.

[00:01:37.78] Joe: But the real magic, the clever part happens next.

[00:01:40.19] Joe: It's not just transcribing one long audio stream.

[00:01:42.78] Joe: It figures out who was speaking.

[00:01:44.19] Joe: That's called speaker diarization.

Joe

[00:01:45.60] Joe: Direwasation. Okay.

Joe

[00:01:46.53] Joe: That's handled by a specific free AI model from Huggieface called PianoSpeaker Diarization.

[00:01:52.41] Joe: This model runs first, slices up the audio based on who's talking, then feeds those speaker

[00:01:56.61] Joe: specific chunks to the transcription AI.

Sally

[00:01:58.68] Sally: Ah, okay, that makes a huge difference compared to just getting a wall of text you have to

[00:02:04.96] Sally: sort out later.

[00:02:05.96] Sally: So for you, the listener running this, what do you actually get on your computer at the

[00:02:09.84] Sally: end?

Joe

[00:02:10.37] Joe: You get two main things, primarily.

[00:02:12.81] Joe: First, there's this really neat interactive HTML file.

[00:02:15.93] Joe: It shows the video player right next to the transcript text.

Sally

[00:02:18.54] Sally: In the cool part, the documentation says you can click a word in the text.

Joe

[00:02:21.85] Joe: Yeah, click any word and bam, the video jumps straight to that exact spot.

[00:02:26.65] Joe: You do need both the HTML file and the original video file together for that to work, obviously.

Sally

Sally

[00:02:26.30] Sally: Hmm.

Sally

[00:02:31.63] Sally: That's incredibly useful for a review.

[00:02:33.72] Sally: But what about for more serious analysis?

Joe

[00:02:36.43] Joe: Right, that's the second output, an automatically generated

[00:02:39.27] Joe: DOCX file, a Word document.

[00:02:42.47] Joe: And this one is structured for analysis.

[00:02:44.61] Joe: It includes the speaker label

[00:02:46.19] Joe: and the precise timestamp for each segment.

[00:02:48.55] Joe: So it's basically ready to go for import into research software

[00:02:51.15] Joe: like MxQDA right out of the box.

Sally

[00:02:53.30] Sally: Okay, that's the game changer, because usually, yeah, you get raw text and spend hours

[00:02:57.48] Sally: syncing it up, marking speakers, having the speaker tracking and the timestamps built

[00:03:01.44] Sally: in automatically, that saves so much time, huge value boost for actual analysis or even

[00:03:07.08] Sally: just editing.

[00:03:08.08] Sally: All right, so the output is powerful, but what about getting it installed?

[00:03:12.84] Sally: You mentioned accessibility being key.

[00:03:15.40] Sally: It's a Python tool, which can sometimes mean tricky setups.

[00:03:18.48] Sally: How do they handle that?

Joe

[00:03:19.48] Joe: Well, they were smart about it, they basically offered two doors, depending on your technical

[00:03:22.88] Joe: comfort level.

Joe

[00:03:23.54] Joe: Two doors. Okay.

Sally

Joe

Joe

[00:03:24.50] Joe: Path number one is the direct Python route.

[00:03:27.26] Joe: If you're comfortable with command lines, terminals,

[00:03:29.82] Joe: package managers, you just use PIP.

[00:03:32.54] Joe: PIP three install, transcribe with whisper.

[00:03:34.98] Joe: Simple enough for Python folks,

[00:03:36.47] Joe: you do also need FFM Peg installed separately.

[00:03:39.50] Joe: That's the standard tool for handling

[00:03:40.94] Joe: audio video conversions, which this relies on.

Sally

[00:03:43.19] Sally: Okay, standard for developers maybe, but what about the person from their mission statement,

[00:03:47.55] Sally: the one who just clicked stuff?

Joe

[00:03:49.62] Joe: Ah, right.

[00:03:50.98] Joe: For them, the strongly recommended path is Docker,

[00:03:54.74] Joe: either Docker Desktop or just the Docker Engine.

[00:03:58.09] Joe: The documentation basically says if you're asking

[00:04:00.38] Joe: what's Python, then Docker is probably your answer.

Sally

[00:04:03.23] Sally: Docker. So you install this one application, Docker desktop, and it kind of sidesteps all the Python

[00:04:08.67] Sally: version issues and dependencies. Is that the idea?

Joe

[00:04:11.82] Joe: That's exactly it.

[00:04:12.86] Joe: It's the secret weapon for ease of use here.

[00:04:15.22] Joe: Instead of you managing all the bits and pieces,

[00:04:17.26] Joe: you run a simple command line instruction.

[00:04:19.74] Joe: That pulls down a pre-packaged Docker container.

[00:04:22.90] Joe: Think of it like a mini computer

[00:04:24.36] Joe: with everything already set up inside.

[00:04:26.14] Joe: It's kept up to date by the developer.

Sally

[00:04:27.85] Sally: And does that give you a graphical interface?

Joe

[00:04:29.91] Joe: Often yes.

[00:04:30.91] Joe: The setup usually includes a web user interface.

[00:04:33.31] Joe: You just open your web browser to a local address

[00:04:35.35] Joe: like http.logalhost.5001.

[00:04:38.75] Joe: And right there in your browser,

[00:04:39.75] Joe: you can upload your file, maybe type in the speaker names

[00:04:42.07] Joe: if you know them and click go.

[00:04:44.03] Joe: It hides all the command line stuff.

Sally

[00:04:45.27] Sally: That sounds much more user-friendly, does there any catch?

Joe

[00:04:48.29] Joe: It's a small one for Windows users.

[00:04:50.49] Joe: If you're on Windows and using Docker, you need to install WSL first.

[00:04:53.97] Joe: That's the Windows subsystem for Linux.

[00:04:56.49] Joe: Docker on Windows relies on WSL to run properly.

Sally

[00:04:58.90] Sally: Okay, WSEL first, then Docker, got it.

[00:05:02.26] Sally: But you mentioned something absolutely required

[00:05:04.34] Sally: regardless of the path.

Joe

[00:05:05.67] Joe: Yes, whether you go Python or Docker, there's one critical step before you can transcribe

[00:05:11.27] Joe: anything with speaker separation.

[00:05:13.59] Joe: You need a hugging face off token.

[00:05:15.87] Joe: It's non-negotiable for the diurization part.

Sally

[00:05:18.36] Sally: hugging face.

[00:05:19.28] Sally: Okay, they host a lot of AI models.

[00:05:21.84] Sally: Why is a token needed for this specific tool

[00:05:24.88] Sally: if it's running locally?

Joe

[00:05:26.56] Joe: Good question.

[00:05:27.72] Joe: It's not actually for the transcription part

[00:05:29.48] Joe: using FastWisper.

[00:05:30.68] Joe: That runs fine offline, no token needed.

[00:05:33.40] Joe: The token is required specifically to download the AI models

[00:05:36.60] Joe: that do the speaker diurization,

[00:05:38.00] Joe: that speaker separation we talked about.

Sally

[00:05:39.35] Sally: Ah, so the piano models need it.

Joe

[00:05:41.01] Joe: Exactly. Those specific models, Piano Speaker Diarrhization 3.1 and Peno segmentation 3.0

[00:05:47.41] Joe: are what they call gated models on hugging face. You need to prove you have permission to

[00:05:50.93] Joe: download them even though they're free. The token acts as your key.

Sally

[00:05:54.49] Sally: Okay, so this is crucial for anyone setting it up.

[00:05:57.61] Sally: Let's walk through the steps based on the docs.

[00:06:00.11] Sally: First, create a free hugging face account, simple enough.

[00:06:04.57] Sally: Then, and this sounds like the step people might miss,

[00:06:07.57] Sally: you have to go to the pages

[00:06:08.89] Sally: for both of those P&O models.

Joe

[00:06:10.17] Joe: Right, both speaker diurization 3.1 and segmentation 3.0.

Sally

[00:06:14.42] Sally: And click the button that says something like, use this model or request access and agree

[00:06:19.54] Sally: to their terms of service.

[00:06:21.50] Sally: The docs warn that if you get a 401 or 403 error later, it means you probably skipped this

[00:06:26.14] Sally: explicit acceptance step.

Joe

[00:06:27.82] Joe: Precisely.

[00:06:28.66] Joe: Once you've accepted the terms for both models,

[00:06:31.04] Joe: you go to your hug and face account settings

[00:06:32.96] Joe: and create an access token.

[00:06:34.20] Joe: Make sure it's a read token.

[00:06:35.50] Joe: It'll be a string of characters starting with HF.

Sally

[00:06:37.50] Sally: And then you have to tell your computer about this token.

Joe

Sally

[00:06:49.77] Sally: Hello.

Sally

[00:07:01.92] Sally: Ah, good tip.

[00:07:03.44] Sally: Set X for permanent on Windows.

[00:07:05.44] Sally: Okay, so token set, permissions granted, then you run the tool.

Joe

[00:07:10.44] Joe: than you were on the tool.

[00:07:11.28] Joe: And just ahead of the very first time you run it,

[00:07:13.24] Joe: my take a little while,

[00:07:14.08] Joe: I have to download those fairly large P&O models

[00:07:16.28] Joe: from hugging face using your token.

[00:07:19.08] Joe: Could be a few minutes depending on your internet.

Sally

[00:07:20.91] Sally: But after the first time, much.

Joe

[00:07:21.77] Joe: much faster.

Sally

Sally

Joe

[00:07:22.64] Joe: Once the models are downloaded, they're cached locally on your machine.

[00:07:26.16] Joe: Subsequent runs just use the local copies, so they should be significantly quicker.

Sally

[00:07:30.69] Sally: Okay, good to know.

[00:07:32.77] Sally: So we covered the main outputs,

[00:07:34.65] Sally: the Interactive HTML and the Analysis Ready DOCX.

[00:07:38.25] Sally: Are there any other details or maybe caveats mentioned

[00:07:42.17] Sally: in the documentation?

Joe

[00:07:43.40] Joe: a few practical points.

[00:07:45.20] Joe: They mentioned the HTML output isn't stuck as HTML.

[00:07:48.96] Joe: There's actually an included helper script,

[00:07:50.96] Joe: HTML to docs, that can convert that interactive HTML file

[00:07:55.68] Joe: into other formats if you need them,

[00:07:57.20] Joe: like ODTE for Libra Office or even PDF.

Sally

[00:08:00.23] Sally: Oh, handy. Flexibility's good.

Joe

[00:08:01.76] Joe: What?

Sally

[00:08:01.88] Sally: What about other standard subtitle formats?

Joe

[00:08:04.14] Joe: It does also generate VTT files, you know the standard web video text tracks format that lots of players use for captains.

Sally

[00:08:09.98] Sally: Right, see your VTTs too.

Joe

[00:08:11.67] Joe: You do, but this is a critical distinction they make.

[00:08:14.43] Joe: The VTT files generated by this tool

[00:08:16.93] Joe: do not contain the speaker information.

[00:08:19.15] Joe: They're just the time text.

[00:08:20.19] Joe: Find for basic subtitles, maybe.

[00:08:22.07] Joe: But totally useless if you need to know who said what,

[00:08:24.51] Joe: which is often the whole point for researcher detailed review.

[00:08:27.39] Joe: For that, the DOC X is the way to go.

Sally

[00:08:29.15] Sally: got it. VTT for basic captions, DOCX for the real analysis with speaker data

[00:08:33.71] Sally: that makes sense. Anything else? Platform quirks.

Joe

[00:08:36.52] Joe: Yeah, and note about Max, if you're running on Apple,

[00:08:39.56] Joe: Silicon, M1, M2, M3, M4 chips,

[00:08:43.28] Joe: the default setup works fine,

[00:08:45.12] Joe: but the performance might be a bit slower

[00:08:46.96] Joe: than you'd expect from such powerful hardware.

[00:08:49.64] Joe: Apparently to really leverage the GPU

[00:08:51.60] Joe: or the neural engine on those Max for faster processing,

[00:08:54.60] Joe: you'd need to install some optional

[00:08:55.96] Joe: more specialized Python packages with Core ML support.

[00:08:59.28] Joe: The basic setup doesn't do that automatically.

Sally

[00:09:01.33] Sally: Okay, so power users on Mac might need an extra step for peak speed.

[00:09:04.97] Sally: What about Windows?

[00:09:06.09] Sally: Aside from the WSL thing for Docker.

Joe

[00:09:08.00] Joe: Yes, this part was quite amusing in the documentation. There's a section on Windows installation.

[00:09:13.75] Joe: And the author includes a note stating very clearly that the Windows instructions were generated by

[00:09:18.63] Joe: chatGPT.

Sally

[00:09:19.69] Sally: Seriously, generated by chat GPT.

Joe

[00:09:21.83] Joe: Yep, and followed by the disclaimer that they are not tested by the author personally.

[00:09:26.27] Joe: Apparently the author's own experience with Windows ended, and I quote, around Windows

Joe

[00:09:31.32] Joe: Wow, okay, points for honesty, I guess.

[00:09:33.24] Joe: That's refreshingly transparent.

Joe

[00:09:33.36] Joe: Yeah.

Joe

[00:09:35.01] Joe: It really is.

[00:09:35.85] Joe: It tells you a front.

[00:09:37.05] Joe: Look, this is primarily built and tested on Linux and Mac.

[00:09:40.13] Joe: The Windows stuff is provided as is.

[00:09:42.13] Joe: Maybe it works, maybe it doesn't.

[00:09:43.53] Joe: We didn't really check.

[00:09:44.81] Joe: You don't often see that level of candor

[00:09:47.13] Joe: in open-source project docs.

[00:09:48.77] Joe: It manages expectations perfectly.

Joe

Sally

[00:09:44.50] Sally: Yeah.

Sally

[00:09:50.62] Sally: Absolutely. Okay, so let's try and synthesize this. What's the big picture here?

Joe

[00:09:54.22] Joe: I think the key takeaway is how this tool manages to sort of democratize some pretty sophisticated

[00:10:00.32] Joe: AI.

[00:10:01.32] Joe: You get high quality transcription from Whisper, plus that really crucial speaker diurization

[00:10:05.96] Joe: layer.

[00:10:06.96] Joe: And it's packaged up especially via Docker in a way that lets people who aren't AI experts

[00:10:12.24] Joe: or Python wizards actually use it.

Sally

[00:10:14.82] Sally: Uh huh.

Joe

[00:10:15.19] Joe: all while keeping the data completely private on their own machine.

Sally

[00:10:18.47] Sally: Right, it bridges the gap between powerful AI and practical usability for privacy conscious

Joe

[00:10:24.32] Joe: Exactly. It takes these complex open-source bits and pieces and integrates them into something genuinely

[00:10:29.76] Joe: useful and accessible for, you know, researchers, journalists, anyone needing private transcription.

Sally

[00:10:34.65] Sally: Okay, so for a final thought, here's something interesting that this whole set of highlights.

[00:10:38.61] Sally: You get this powerful, local, privacy-preserving AI tool, fantastic.

[00:10:44.25] Sally: But you can only unlock its full potential, specifically that speaker separation, after you

[00:10:49.17] Sally: perform this little digital handshake with hugging face.

[00:10:51.69] Sally: You need that free account, you need to accept terms, you get that token.

[00:10:54.29] Sally: So the only real gatekeeper to running this advanced self-hosted AI isn't technical skill

[00:10:59.37] Sally: anymore, thanks to Docker, just having this free online account.

[00:11:03.53] Sally: And agreeing to the terms for the models.

[00:11:05.93] Sally: What does that imply for the future?

[00:11:08.01] Sally: As more powerful, specialized AI models become available, but maybe gated behind similar,

[00:11:13.37] Sally: simple, free, but required accounts.

[00:11:15.33] Sally: What does that mean for access, for control, even for privacy in the long run?

[00:11:18.65] Sally: That's definitely something interesting for you to mull over.

Joe

[00:11:00.72] Joe: You

Joe