

1 August 2026

CapsQual: GUI Tool for Turning ASR Output into Qualitative Transcripts

Anouâr Gadermann^{1, 2} 

¹ Qingdao University of Science and Technology ² Paderborn University

Abstract

CapsQual is an open-source CLI/GUI workstation for transforming subtitle files created through automatic speech recognition (ASR) into qualitative transcripts following established research standards. Users can quickly assign speakers to segments, import and synchronize audio files, adjust timestamp information using an intuitive waveform widget, enter transcription symbols and sequentially edit and refine transcripts, thus aiding in their preparation for analysis within or without computer-assisted qualitative data analysis software (CAQDAS). The export function implements basic formatting templates, while offering ample options for customization such as custom timestamp format, adjustable line-wrapping and concatenation of speaker turns. The software facilitates a reflective computer-assisted workflow tailored to the properties of ASR-generated data, thus bridging the gap between ASR tools and highly sophisticated transcription software.

Keywords: transcription, open source software, automatic speech recognition, qualitative data

1. Introduction

Over a decade ago, Dresing and Pehl (2015) commented on the notion of applying speech recognition software to qualitative interview transcription: “Leider, leider. . . Es geht noch nicht.” (Roughly: *What a pity. . . it can’t be done yet.*) Nowadays, automatic transcription of audio data is available to anyone who has access to a computer with sufficient memory and a decent CPU (though higher-end hardware, especially a fast graphics processor, does result in faster transcription) (Haberl et al. 2023). The ongoing rise in data privacy awareness has aided in the legitimization of localized ASR models for transcription of sensitive interview data (Wollin-Giering et al. 2024). Moreover, the advent of ASR-supported transcription workflows resulted in a substantial reduction of manual labor, since qualitative researchers no longer have to type in utterances key by key (Haberl et al. 2023).

However, while current trends in transcription seem to be pointing towards more automation and less intervention, researchers largely agree that manual transcription has several

benefits for qualitative research: The process involves interpretative actions and decisions which influence the way in which data reduction occurs, thus shaping researchers' perspectives (Dresing and Pehl 2020; Przyborski and Wohlrab-Sahr 2021). Broadly, it results in more familiarity with the material (Dörnyei 2007). Unfortunately, manual transcription remains labor-intensive and professional software for manual transcription such as EXMARaLDA (Schmidt and Wörner 2014) has a steep learning curve, which, to novice researchers dealing with transcription marginally (e.g., those working with QUANT-qual mixed-method designs while on a deadline), might prove mildly intimidating. On the other hand, fully automated open-source solutions such as noScribe (Dröge 2025) and aTrain (Haberl et al. 2023) put more focus on taking the burden of producing an initial transcript off the researcher's shoulder, while at the same time leaving the manual editing (especially regarding different transcript formatting conventions) much up to the researcher's skill, using traditional means.

CapsQual (Gadermann 2026) aims to fill this niche: Unlike traditionalist transcription software which might not build on recent developments in speech recognition, nor like all-in-one solutions which focus on the automatic production of transcripts from audio material, CapsQual is designed to assist in the process of sequentially preparing ASR output for different forms of qualitative data analysis such as content, conversation or reconstructive analysis.

2. State of the field

Many server-based providers of automated transcription services such as TurboScribe (Foged 2023) have appeared over the past few years. Most of them build on existing ASR models such as Whisper (OpenAI 2023). Some distributors of proprietary CAQDAS, such as MAXQDA (Software 2024), also offer cloud-based services to transcribe qualitative data for analysis. NoScribe (Dröge 2025) and aTrain (Haberl et al. 2023) are freely available open-source programs that implement offline speech recognition models to produce diarized transcripts following standardized conventions common in qualitative and thematic analysis. Notably, noScribe also includes an editor for manually reviewing transcripts. GailBot (Umair et al. 2022) is an API-based software package that combines speech-to-text systems and various plugins to automatically generate Jeffersonian transcripts from audio.

In linguistics, highly sophisticated, specialized transcription work suites have become the established norm. EXMARaLDA (Schmidt and Wörner 2014) is a system of programs aimed at producing accurate transcriptions from video or audio material that, among others, display multiple transcript tiers in *Partitur*-layouts resembling musical scores. Similarly, ELAN is a widely-used software suite for multimodal research developed by the Max Planck Institute for Psycholinguistics (2026). It offers a large variety of annotation and export options. PRAAT (Boersma et al. 2026) is a comprehensive software system commonly used in phonetic research. It offers tools such as spectrographic, intensity and pitch analysis.

Few tools exist to help researchers adapt ASR output to their own research needs without extensive training. The web app Transcript Timestamp Wizard (Sponholz and Mueller 2024; see also Sponholz et al. 2025) converts timestamps created by Adobe Premiere Pro to a CAQDAS-compatible format. On the whole, there is a void between software solutions for automated transcription of audio data and those for advanced linguistic annotation. No dedicated offline programs are known for assisting in the semi-manual preparation of specialized transcripts (such as GAT2 or TiQ) from ASR output.

3. Software design

What started as a simple AI-generated Python script for reformatting timestamps, quickly grew into a versatile transcription GUI workstation, capable of transforming common subtitle files into transcripts following established transcription conventions such as GAT2 (Selting et al. 2011), TiQ (Przyborski and Wohlrab-Sahr 2021) as well as Kuckartz and Rädiker (2024) or Dresing and Pehl (2020). Within the course of a year, new functions were implemented to simplify the process of semi-manual transcript preparation. Besides subtitle files (CapsQual currently supports .SRT, .VTT, .TSV, .JSON and .TXT formats), users can import audio files into projects, synchronize audio and transcript, adjust playback speed, jump to positions in audio from the transcript (and vice versa). Comments, pauses and other transcription symbols can be chosen from a symbols dialog with tabs for different conventions. Custom symbols can also be configured, saved and loaded. Besides the sequential editing, the program offers automated modification functions: stripping punctuation, converting to lowercase and calculating pauses for segments without voice activity. The final result can be exported to .HTML, .DOCX, .TXT or .SRT format.

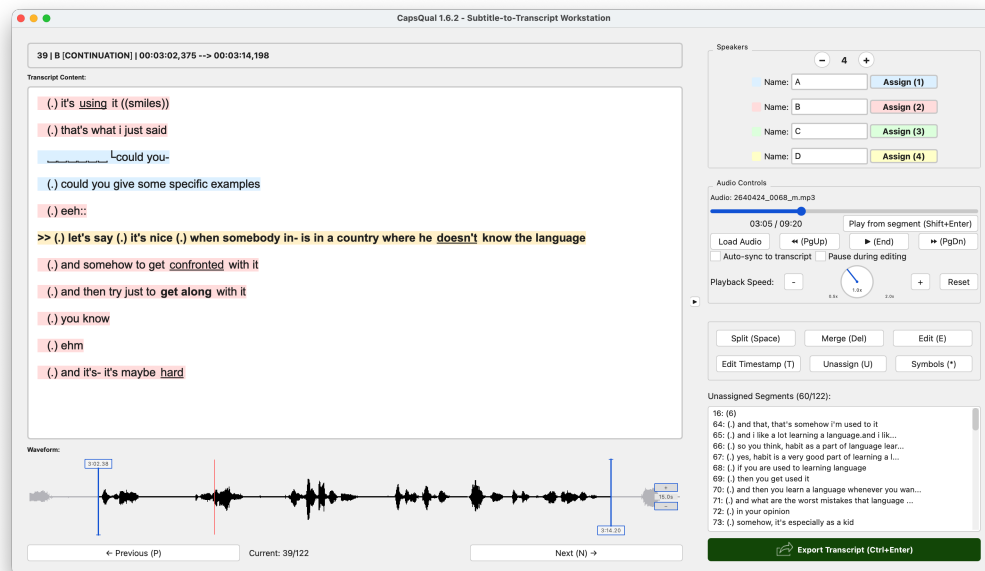


Figure 1: Editing transcript content in CapsQual 1.6.2

CapsQual was developed in Python. The CLI, parsers and transcript generators use pure Python, while the GUI was built using the Qt application framework (PyQt5). The audio features make use of soundfile, numpy as well as the python-vlc bindings and VLC libraries¹. Python-docx is used to realize transcript export to .DOCX format. Agentic AI was used to implement the required functions, following an approach that (Prevett 2026) calls “theory-operationalising tool co-creation”.

¹As of version 1.6.2, VLC needs to be installed on the system for audio capabilities, whereas future releases might bundle the VLC libraries directly.

Special attention was given to keyboard accessibility. All major editor features have shortcut bindings, most of which are displayed on the UI buttons. CapsQual is cross-platform compatible and the GUI has been optimized to run on computers with a resolution of at least 800*600. Binaries are released for Windows, macOS and Linux (Ubuntu/Debian).

4. Usage example

To outline the sequential transcription logic behind CapsQual and to further illustrate its functions, an exemplary workflow is presented here:

- 1) **Import subtitle (and optionally audio) files.** Subtitle files can be imported² using the main menu, via drag and drop or by clicking the `Load Audio` button in audio controls. To obtain the best results, importing files containing timestamps (such as .SRT or .VTT) is strongly recommended. If importing an audio file first, the program searches for corresponding subtitle files in the same folder and asks whether to import them.
- 2) **Assign speakers.** Once the files are imported, speakers need to be assigned to the segments. This is done easily by using the number keys or UI buttons (see [Figure 1](#)). Once a segment is assigned, the selection moves to the next segment, thus speeding up the process significantly. In many cases (e.g., in interviews with two easily discernible speakers), this step can be completed without audio confirmation or by listening only to segments where the speaker is not clear at once. Sometimes ASR tools erroneously merge talk by multiple speakers into one segment. To resolve this, sections can be split by pressing space before assigning speakers.
- 3) **Review the content and refine segments.** Before any major reformatting takes place, the segments need to be manually reviewed for errors. This depends largely on the accuracy of the material provided (which, in turn, depends on the ASR model as well as the quality of the audio material). Syncing the audio playback and segment selection (`Ctrl+l`), jumping to the current audio location (`l`) and adjusting playback speed (`+` and `-`) may come in useful. Line content can be edited by pressing `e`. For conventions where transcript lines are based on special phrasing such as intonation phrases (i.e., in conversation analysis), special attention should be given to the accuracy of the segmentation. By splitting (`space`) and merging (`del`) the segments, as well as by adjusting the start and end times of the timestamps by dragging the handles in the waveform viewer or by pressing `t`, the segmentation accuracy can be improved. If the convention used does not require transcript lines based on phrasing, this partial step can be skipped, since the segments are eventually concatenated into speaker turns.
- 4) **Add overlaps, comments, pauses and other symbols.** Annotating the content remains the most labor-intensive part of the transcription process, though the scope depends largely on the requirements and purpose of the transcript. Imported transcript text can be stripped of punctuation marks or converted to all lowercase automatically, if required. Calculated pauses for parts with no voice activity can be added. These functions can be accessed from the menu bar (`Edit → Modify transcript`). The

²Automatic speech recognition is *not* integrated in CapsQual, since the software is designed to deal with existing ASR output. For more information on ASR tools, see Wollin-Giering et al. (2024) or [State of the field](#).

remaining process involves sequentially moving through the segments using left and right cursors, adding transcription symbols (* to open the symbols dialog or . to add micropauses).

- 5) **Finalize and export.** After correcting the transcript, checking for overlooked errors and inconsistencies is generally advisable. Finally, the transcript should be configured for final export in the export preview dialog (Ctrl+Enter). The required format can be set up by choosing a basic transcription convention and customizing timestamps, concatenation, line-wrapping and other options. The file can then be exported in the required file format (see [Figure 2](#)).

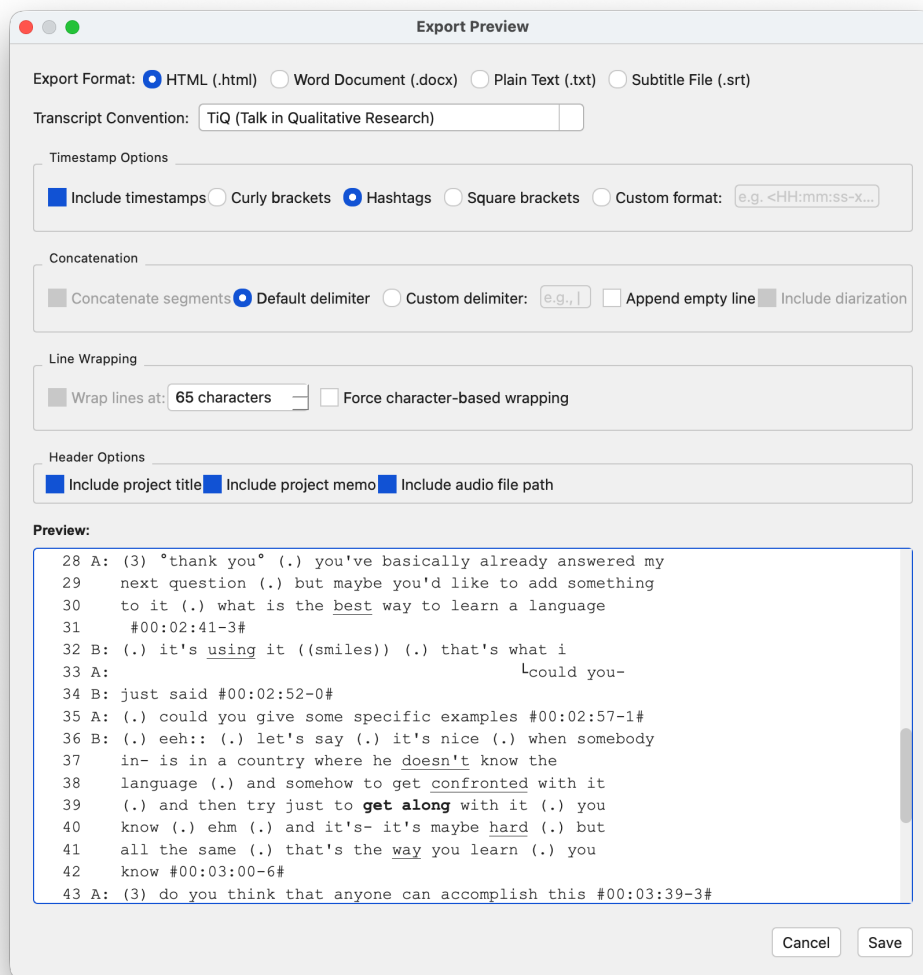


Figure 2: Exporting a transcript in CapsQual 1.6.2

The suggested workflow is not meant as a strict set of rules that have to be followed step by step, but an example of the basic usage logic in CapsQual. In actuality, data preparation hardly ever follows a linear path. It may include content revisions, adjustments to the choice of used symbols or previews of the transcript in different formats. Frequently saving and keeping the CapsQual project for future revision is highly recommended.

5. Conclusion and future outlook

CapsQual bridges the gap between ASR models and complex transcription work suites and simplifies the preparation of qualitative data for work in CAQDAS. The proposed workflow represents a compromise between the convenience of AI transcription and precision associated with advanced analytical software. The open-source tool provides a freely available and user-friendly means of preparing qualitative data for analysis efficiently while retaining full autonomy over the process, allowing users to reflect on transcription conventions and become familiar with their qualitative data.

While CapsQual grew out of research needs related to second language acquisition and motivation for learning German (e.g., [Gadermann in print](#)), its potential for application spans across various fields and disciplines and encompasses integration into other projects such as open-source CAQDAS and ASR tools. The rapid development of artificial intelligence raises the question whether research software projects such as CapsQual will remain relevant for the scientific community in the future or represent merely brief artifacts in time, quickly outdated and forgotten. Either way, the implementation of methodological principles into software may be viewed not only as a means to an end, but also as an end in itself – making the tool becomes a way of learning the craft.

6. Acknowledgements

Special thanks go to Chahra Gadermann for the interview used in the screenshots, to Franz-J. Gadermann for proof-reading this article multiple times and to Rod Khlystun for testing the interface on macOS during an early development stage. I would also like to acknowledge the developers of the PyCharm IDE ([JetBrains s.r.o 2026](#)), the Refact agentic coding plugin ([Vakhreev 2026](#)) and DeepSeek AI ([DeepSeek AI 2026](#)), which helped with drafting, refactoring and debugging during the development process.

References

- Boersma, Paul, David Weenink, and Anastasia Shchupak. 2026. *Praat: Doing Phonetics by Computer*. Computer software. <https://praat.org>.
- DeepSeek AI. 2026. *DeepSeek-V4*. Large language model. <https://www.deepseek.com/>.
- Dörnyei, Zoltán. 2007. *Research Methods in Applied Linguistics*. Oxford University Press.
- Dresing, Thorsten, and Thorsten Pehl. 2015. *Praxisbuch Interview, Transkription & Analyse*. Self-published. https://doi.org/10.1007/978-3-658-18387-5_56-2.
- Dresing, Thorsten, and Thorsten Pehl. 2020. “Transkription.” In *Handbuch Qualitative Forschung in der Psychologie*, edited by Günter Mey and Katja Mruck. Springer Fachmedien Wiesbaden. https://doi.org/10.1007/978-3-658-18387-5_56-2.

- Dröge, Kai. 2025. “noScribe. AI-powered Audio Transcription.” In *GitHub Repository*. Computer software. <https://github.com/kaixxx/noScribe>.
- Foged, Leif. 2023. *TurboScribe: Transcribe Audio and Video to Text*. Online tool. <https://turboscribe.ai/>.
- Gadermann, Anouâr. 2026. *CapsQual: Subtitle-to-Transcript Workstation*. Version 1.6.2. Computer software. <https://doi.org/10.5281/zenodo.21769829>.
- Gadermann, Anouâr. in print. *Motivationsprofile als passiv wahrgenommener Deutschlerner in chinesisch-deutschen MINT-Studiengängen am Beispiel der CDTF*.
- Haberl, Armin, Jürgen Fleiß, Dominik Kowald, and Stefan Thalmann. 2023. *Take the aTrain. Introducing an Interface for the Accessible Transcription of Interviews*. <https://doi.org/10.48550/arXiv.2310.11967>.
- JetBrains s.r.o. 2026. *PyCharm 2026.1.4*. Computer software. <https://www.jetbrains.com/pycharm/>.
- Kuckartz, Udo, and Stefan Rädiker. 2024. *Qualitative Inhaltsanalyse. Methoden, Praxis, Umsetzung mit Software und künstlicher Intelligenz*. Beltz Juventa.
- Max Planck Institute for Psycholinguistics. 2026. *ELAN*. Computer software; Nijmegen: Max Planck Institute for Psycholinguistics, The Language Archive. <https://archive.mpi.nl/tla/elan>.
- OpenAI. 2023. “Whisper.” In *GitHub Repository*. Computer software. <https://github.com/OpenAi/Whisper>.
- Prevett, Pauline. 2026. “Theory-Operationalising Tool Co-Creation: Building Methodology-Specific Analysis Tools Through Human-AI Collaboration. A Position Statement for Qualitative Research Methodology.” *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research* 27. <https://doi.org/10.17169/fqs-27.2.4596>.
- Przyborski, Aglaja, and Monika Wohlrab-Sahr. 2021. *Qualitative Sozialforschung: Ein Arbeitsbuch*. 5. Auflage. De Gruyter Oldenbourg.
- Schmidt, Thomas, and Kai Wörner. 2014. “Exmaralda.” In *Handbook on Corpus Phonology*. Oxford University Press. <https://www.exmaralda.org>.
- Selting, Margret, Peter Auer, Dagmar Barth-Weingarten, et al. 2011. “A system for transcribing talk-in-interaction: GAT 2 translated and adapted for English by Elizabeth Couper-Kuhlen and Dagmar Barth-Weingarten.” *Gesprächsforschung* 12 (April): 1–51.
- Software, VERBI. 2024. *MAXQDA 2024*. Computer software. <https://www.maxqda.com>.
- Sponholz, Jakob, and Maximilian Mueller. 2024. *Transcript Timestamp Wizard*. Computer software. <https://ttw.digitalemedienbildung.de/>.
- Sponholz, Jakob, Andreas Weilinghoff, and Juliane Schopf. 2025. *Halving transcription time: A fast, user-friendly and GDPR-compliant workflow to create AI-assisted transcripts for content analysis*. <https://doi.org/10.48550/arXiv.2503.13031>.
- Umair, Muhammad, Julia Beret Mertens, Saul Albert, and Jan P. de Ruiter. 2022. “GailBot: An automatic transcription system for Conversation Analysis.” *Dialogue & Discourse* 13 (1): 63–95.
- Vakhreev, Sergey. 2026. “Refact.” In *GitHub Repository*. Computer software. <https://github.com/JegernOUTT/refact>.
- Wollin-Giering, Susanne, Markus Hoffmann, Jonas Höfting, and Carla Ventzke. 2024. “Au-

tomatic Transcription of English and German Qualitative Interviews.” *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research* 25. <https://doi.org/10.17169/fqs-25.1.4129>.