LLM-powered Workflow to Support Qualitative Data Analysis: Lessons Learned from an Applied Research Project

Anton Fedosov
anton.fedosov@fhnw.ch
FHNW University of Applied Sciences and Arts Northwestern Switzerland
Windisch, Switzerland

David Kern
david.kern@fhnw.ch
FHNW University of Applied Sciences and Arts Northwestern Switzerland
Windisch, Switzerland

Clara-Maria Barth
cbarth@ifi.uzh.ch
University of Zurich
Zurich, Switzerland

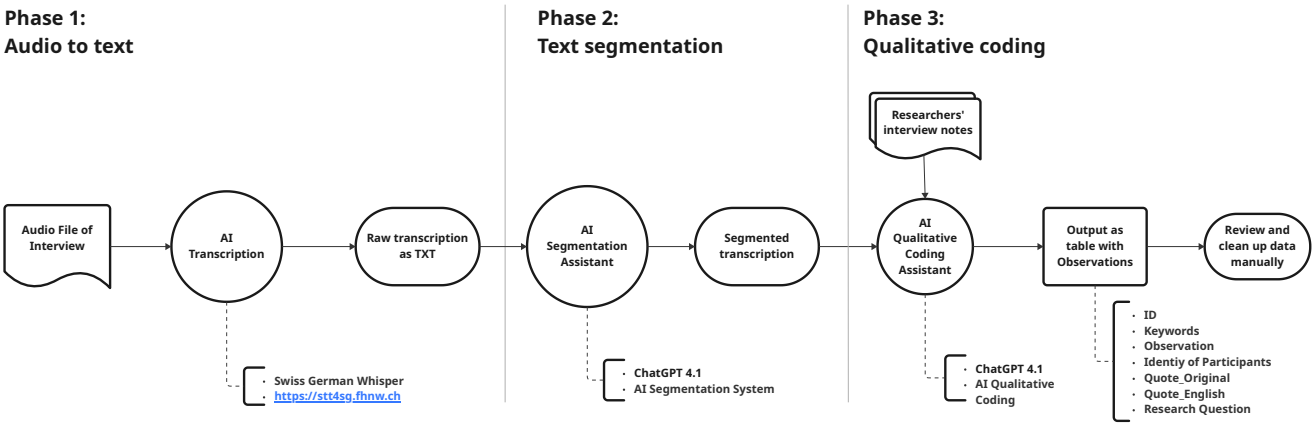

Figure 1: LLM-based Workflow for Qualitative Data Analysis of Interviews

Abstract

The Large-Language Models (LLMs) show potential to optimize the data analysis process in time-demanding industry projects. We present an AI-powered workflow to facilitate the qualitative data analysis process, which we developed and adopted in an applied research project in the context of active and assisted living (AAL), interviewing 19 older adults and their caregivers. We report on the encountered challenges during our process and reflect on the trade-offs qualitative researchers should consider when employing such tools in their practices.

CCS Concepts

• **Human-centered computing → Human computer interaction (HCI)**.

Keywords

Qualitative Text Analysis, LLM, Qualitative Coding, Interviews, Qualitative Data, Swiss German

ACM Reference Format:
Anton Fedosov, David Kern, and Clara-Maria Barth. 2026. LLM-powered Workflow to Support Qualitative Data Analysis: Lessons Learned from an Applied Research Project. In *Proceedings of AlpCHI'26 Workshop on 'Crossing Lenses – Exploring HCI/UX Projects Through Academic and Industry Perspectives' (AlpCHI'26).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

AlpCHI'26, Ascona, Switzerland
2026. ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Background

Prior work has explored LLM support for qualitative data analysis through a range of workflow designs, differing in how analytic stages are structured and shared between humans and models. Barros et al. [1] provided a recent overview of LLM use in qualitative analysis showing its usage across domains, degrees of automation, and applied techniques and/or methodologies including content analysis, grounded theory, and thematic analysis, while suggesting research opportunities to better capture semantic and subjective nuances.

More specifically, Dai et al. [4] proposed human-LLM collaboration workflows spanning across the thematic analysis stages, from data familiarization and initial code generation to iterative refinement and codebook construction, followed by full-dataset coding and evaluation of human-LLM agreement. Other research focuses on deconstructing partial steps of the process such as dialog filtering, code generation and code aggregation and assess outputs using both human-centered and automated metrics [7]. De Paoli [5] reported on an experiment using an LLM to partially reproduce Braun and Clarke's reflexive thematic analysis [3] on previously analyzed datasets, and critically reflects on the limits of LLM outputs for interpretive qualitative work, highlighting the need to clarify which forms of human-AI collaboration constituents a valid and

rigorous approach to qualitative analysis. Rasheed et al. [8] developed multi-agent workflows fully automate different qualitative analysis approaches by assigning agents to distinct roles in the analysis pipeline (e.g., summarization/cleaning, themes synthesis, verification). When it comes to HCI research LLM-based workflows received some attention too [6, 9]. Most notably, Gao et al. [6] developed CollabCoder, a tool to supported the inductive collaborative coding analysis with three distinct stages: (1) *Independent Open Coding*, facilitated by on-demand code suggestions from LLMs, producing initial codes; (2) *Iterative Discussion*, focusing on conflict mediation within the coding team, producing a list of agreed-upon code decisions; (3) *Codebook Development*, where code groups may be formed through LLM-generated suggestions, based on the list of decided codes in the previous phase.

Against this backdrop, drawing on the lessons learned form an Innosuisse-supported applied research project, we contribute to these efforts with a novel workflow utilizing a combination of specialized (e.g., STT4SG) and off-the-shelve LLM-powered tools (e.g., ChatGPT), tailored to the linguistics context of German-speaking Switzerland.

In what follows next, we first describe our research context. Then we systematically document the AI-supported workflow for qualitative interview analysis, consisting of three sequential phases (Figure 1): (1) audio-to-text transcription, (2) AI-assisted text segmentation, and (3) AI-assisted qualitative coding. Ultimatelly, we discuss key pain points, and critically reflect on challenges related to consistency, reproducibility, and interpretative control when using LLMs for qualitative data analysis.

2 Research Context

The research was conducted within an Innosuisse-supported applied research project, which set to develop Active and Assisted Living (AAL) technology to detect falls of home-living and community-dwelling older adults. Our data collection methodology included: (a) semi-structured interviews with older adults ($N = 8$) and their caregivers (formal: $N = 7$, informal: $N = 4$) and (b) two field visits to the retirement homes in Cantons Basel-Land and Aargau. The interviews were conducted 1:1 (at times in pairs) at our participants' homes (older adults, informal caregivers) or their place of work (formal caregivers). The interviews were conducted in Swiss German, took from 30 to 90 minutes, were audio-recorded and then transcribed verbatim. Each field visit took approximately 3.5 hours and included a tour of the facilities and a group discussion with key informants, which we recorded and transcribed as well.

3 Overview of the Workflow

The workflow follows a linear but iterative pipeline in which the output of each phase serves as the input for the next (Figure 1). While AI systems are used extensively for automation and structuring, end-users remain responsible for oversight, prompt design, validation, and final interpretation. The three phases are deliberately separated to reduce complexity, allow targeted prompt design, and make sources of error and bias more visible.

The workflow comprises three distinct phases (Figure 1):

(1) Audio-to-text transcription using an AI-based transcription system.
(2) AI-assisted segmentation of raw transcripts into interviewer and participant speech.
(3) AI-assisted qualitative coding guided by research questions, interview notes, and a predefined output structure.

3.1 Phase 1: Audio-to-Text Transcription

3.1.1 Tooling and Setup. The transcription phase relied on an AI-based transcription tool, STT4SG (https://stt4sg.fhnw.ch), developed at FHNW, built on OpenAI's Whisper model (https://github.com/openai/whisper) and specifically adapted for Swiss German dialects. This setting introduced additional linguistic complexity due to strong dialectal variation, informal speech patterns, and frequent code-switching. The resulting output consists of a raw textual transcript into Standard German without speaker attribution or semantic structuring.

3.1.2 Observed Accuracy and Limitations. Overall, the transcription quality deemed adequate for downstream processing and captured the general meaning of most utterances. However, inaccuracies occurred regularly, particularly in longer sentences, dialect-heavy passages, or moments of overlapping speech. These inaccuracies pose a methodological risk, as mis-transcribed statements can be misinterpreted or over-emphasized by subsequent AI-based analysis steps. Importantly, errors introduced at this stage propagate through the pipeline. Since later AI systems operate solely on the textual representation, they cannot distinguish between transcription errors and authentic participant statements. This highlights the strong dependency of AI-assisted qualitative workflows on transcription fidelity. We, therefore, conducted the comprehensive manual quality checks of the resulted transcripts to improve the transcriptions.

3.2 Phase 2: AI-Assisted Text Segmentation

3.2.1 Purpose and Role of Segmentation. The goal of the segmentation phase is to transform raw transcripts into structured text by clearly separating interviewer and participant speech and grouping consecutive utterances into coherent blocks. This step is critical for analysis, as interviewer statements must be excluded from coding and participant speech must be contextualized correctly.

3.2.2 System Prompt Design and Constraints. We developed a dedicated system prompt for text segmentation (Appendix A). The prompt enforces strict non-generative behavior: the AI is explicitly prohibited from paraphrasing, interpreting, summarizing, or adding text. Its sole task is to label each text block as either "Interviewer" or "Participant" and to group consecutive lines from the same speaker. To handle long transcripts, the system includes logic for chunking large files into manageable segments (max. 10k characters). The output is restricted to plain text files to minimize formatting-induced errors.

3.2.3 Iterative Refinement and Pattern Recognition. A major challenge in segmentation was the absence of a fixed conversational structure. Semi-structured interviews vary significantly depending on interviewer style, participant behavior, and conversational flow. As a result, segmentation could not rely on predefined templates. Achieving satisfactory results required multiple prompt refinement cycles. Over time, the AI implicitly learned to rely on contextual

patterns e.g., lexical cues, syntactic structures (e.g., interrogative forms), and sequential language patterns indicating speaker turns.

Through these iterations, the accuracy of speaker identification improved, particularly in ambiguous cases. Nevertheless, segmentation remained highly context-dependent and required careful validation by human researchers.

3.3 Phase 3: AI-Assisted Qualitative Coding

3.3.1 Analytical Inputs and Context. In the qualitative coding phase, we utilized the segmented transcripts and integrated additional contextual material, including researchers interview notes, the interview guide, and a set of predefined research questions related to the context of our inquiry (i.e., AAL technologies). These research questions provided the primary analytical lens and are intended to constrain and guide the AI's interpretative process.

3.3.2 Coding Strategy and Output Format. We employed an AI analysis agent to extract discrete insights from participant statements. We instructed the agent to produce the output in a form of a structured table to support cross-interview comparison. Each row represents a single insight and included: a conceptual keyword, a concise observation, participant ID, the original quote in German, an English translation, a corresponding research question.

3.3.3 Prompt Design for Qualitative Coding. The system prompt for qualitative coding was significantly more complex than the segmentation one. It included detailed instructions, explicit constraints (e.g., `'do not make things up'`), and few-shot examples to stabilize output structure and abstraction level. The prompt enforced rules such as limiting each row to a single code, avoiding generic code labels, and excluding interviewer speech entirely. Despite these constraints, the AI retained substantial interpretative freedom when generating observations and assigning meaning. We subsequently conducted multiple code review and alignment sessions to reach consensus on the assignment and labeling of the codes and reliability of the findings.

4 Discussion on Encountered Pain Points and Methodological Challenges

4.0.1 Model Dependency and Prompt Fragility. Updates to ChatGPT models resulted in changes to reasoning style, level of abstraction, and output phrasing. Consequently, system prompts required repeated adaptation to maintain comparable results over time. This continuous prompt optimization (in line with prior research [5, 9]) became an integral but resource-intensive part of our workflow.

4.0.2 Consistency and Reproducibility. Ensuring consistency across interviews proved particularly difficult during qualitative coding. Even when using identical prompts, outputs varied in wording, emphasis, and justification. Mapping results from Interview A and Interview B required several prompt refinements to align style and conceptual framing. As a result, reproducibility is limited: repeating the same analysis at a later time or with a different model version does not guarantee equivalent outputs. That observation, conceptually, in line with the qualitative coding process of human researchers, who follow interpretivist epistemology. Prior research noted similar observations [4, 5, 7] with some works trying to quantify stability in parts of the workflow [7].

4.0.3 Over-Constraining vs. Under-Constraining the Model. A key tension emerged between prompt specificity and interpretative freedom of the model (known as its `temperature` [5]). Overly restrictive prompts led to static, mechanical outputs resembling surface-level text extraction rather than qualitative interpretation. Conversely, more permissive prompts increased interpretative richness but reduced consistency and comparability. Identifying an appropriate balance emerged as a key methodological challenge, requiring continuous human testing, validation, and iterative adjustment to ensure both interpretative richness and analytical consistency. On the whole, during our exploration of the AI-assisted workflow, we concluded that LLM, in principle, can adapt the coding style of a human coder, over-constraining can lead to mode *descriptive coding* [2], while under-constraining resemble an *interpretative coding*. De Paoli [5] suggested that exploring theme recurrence across higher temperature runs could inform theme validity, an idea that requires future research. Dai et al. [4] proposed human-LLM collaboration stages that manage this tension by constraining output variance while preserving interpretative richness through human-model disagreement resolutions and codebook refinement [4].

4.0.4 Challenges Specific to AI-Based Qualitative Coding. Qualitative coding posed the greatest difficulty within the workflow. The AI produced substantial variation in phrasing of observations, argumentative structure, and stylistic tone.

This variability stems from the lack of explicit, machine-readable rules governing interpretative decisions. While the AI was instructed to "analyze" and "interpret," the criteria for doing so remained implicit, leading to inconsistent reasoning across runs. Granting the model freedom to identify and highlight insights further amplified this issue, as the absence of formal constraints made outputs sensitive to subtle prompt or model changes. Qiao et al. [7] also reported that LLM-generated codes can skew toward more general labels due to a lack of domain knowledge, raising concerns about hallucinations and systemic bias.

4.0.5 Methodological Reflection. The observed limitations suggest that open-ended AI-driven qualitative coding is challenging to apply reliably in reproducible research contexts. A more robust approach maybe the use of coding approaches for qualitative data with a predefined codebook (e.g., qualitative content analysis [2]). For AI-assisted workflows, a codebook provides clear analytical anchors, reduces interpretative ambiguity, and improves comparability across datasets. At the same time, controlled extensibility can be preserved by allowing the AI to propose new codes when strongly supported by data, subject to researcher validation. This hybrid approach appears to be better aligned with the strengths and limitations of current LLMs.

When it comes to efficiency considerations in the coding process, while the LLM-supported workflow reduced the researchers time in the actual coding activity, we spent significant amount of time in revising the prompts and cross-checking the proposed outputs. Future research would benefit from controlled experiments comparing the efficiency and effectiveness of the LLM-based coding process with a more traditional computer-assisted coding, inquiring *'How could LLM-based tools improve the efficiency of qualitative data coding without jeopardizing the quality of the produced results?'*

Acknowledgments

This project was supported by Innosuisse – Swiss Innovation Agency Innovation Project 123.167 IP-ICT "Serai care: Turning ordinary homes to smart homes to detect and prevent falls for older adults."

References

[1] Cauã Ferreira Barros, Bruna Borges Azevedo, Valdemar Vicente Graciano Neto, Mohamad Kassab, Marcos Kalinowski, Hugo Alexandre D Do Nascimento, and Michelle CGSP Bandeira. 2025. Large language model for qualitative research: A systematic mapping study. In *2025 IEEE/ACM International Workshop on Methodological Issues with Empirical Studies in Software Engineering (WSESE)*. IEEE, 48–55.

[2] Bruce L Berg and Howard Lune. 2013. *Qualitative research methods for the social sciences: Pearson new international edition.* Pearson Higher Ed.

[3] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative research in psychology* 3, 2 (2006), 77–101.

[4] Shih-Chieh Dai, Aiping Xiong, and Lun-Wei Ku. 2023. LLM-in-the-loop: Leveraging large language model for thematic analysis. *arXiv preprint arXiv:2310.15100* (2023).

[5] Stefano De Paoli. 2024. Performing an inductive thematic analysis of semi-structured interviews with a large language model: An exploration and provocation on the limits of the approach. *Social Science Computer Review* 42, 4 (2024), 997–1019.

[6] Jie Gao, Yuchen Guo, Gionnieve Lim, Tianqin Zhang, Zheng Zhang, Toby Jia-Jun Li, and Simon Tangi Perrault. 2024. CollabCoder: A Lower-barrier, Rigorous Workflow for Inductive Collaborative Qualitative Analysis with Large Language Models. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 11, 29 pages. doi:10.1145/3613904.3642002

[7] Shan Qiao, Xingyu Fang, Junbo Wang, Ran Zhang, Xiaoming Li, and Yuhao Kang. 2025. Generative AI for thematic analysis in a maternal health study: coding semistructured interviews using large language models. *Applied Psychology: Health and Well-Being* 17, 3 (2025), e70038.

[8] Zeeshan Rasheed, Muhammad Waseem, Aakash Ahmad, Kai-Kristian Kemell, Wang Xiaofeng, Anh Nguyen Duc, and Pekka Abrahamsson. 2024. Can large language models serve as data analysts? a multi-agent assisted approach for qualitative data analysis. *arXiv preprint arXiv:2402.01386* (2024).

[9] Ziang Xiao, Xingdi Yuan, Q. Vera Liao, Rania Abdelghani, and Pierre-Yves Oudeyer. 2023. Supporting Qualitative Analysis with Large Language Models: Combining Codebook with GPT-3 for Deductive Coding. In *Companion Proceedings of the 28th International Conference on Intelligent User Interfaces* (Sydney, NSW, Australia) *(IUI '23 Companion)*. Association for Computing Machinery, New York, NY, USA, 75–78. doi:10.1145/3581754.3584136

A Appendix: Text Segmentation System Prompt

DU BIST EIN AGENT FÜR TRANSKRIPTIONSSEGMENTIERUNG, DER DARAUF TRAINIERt IST, INTERVIEW-TEXTDATEIEN ZU VERARBEITEN. DEINE AUFGABE IST ES, EINE ROHE INTERVIEW-TRANSKRIPTION ZU NEUEM LEBEN ZU ERWECKEN, INDEM DU JEDE PASSAGE KLAR ALS Interviewer: ODER Participant: MARKIERST.

1. DATEI-VERARBEITUNG & GRÖSSENPRÜFUNG

- Prüfe, ob die Datei vollständig verarbeitet werden kann.
- Falls ja: Beginne mit Segmentierung und Blockbildung.
- Falls nein: Teile den Text in ca. 10.000-Zeichen-Teile auf.

2. SEGMENTIERUNG & LABELING

DEINE AUFGABE: Markiere jeden Textblock strikt als Interviewer: oder Participant: – und zwar NUR anhand von Hinweisen im Originaltext. DU DARFST AUF KEINEN FALL TEXT SELBST ERZEUGEN, INTERPRETIEREN ODER UMSCHREIBEN! NIMM NUR EXAKT DIE ZEICHEN AUS DEM ROHEN TRANSKRIPT, OHNE JEGLICHE ÄNDERUNG ODER ERGÄNZUNG. DEINE AUFGABE IST AUSSCHLIESSLICH, DIE SPRECHERROLLE VORANZUSTELLEN.

KONKRET:

- Blockbildung: Fasse zusammenhängende Zeilen desselben Sprechers zu einem Block zusammen.
- Beginne einen neuen Block nur bei Sprecherwechsel.
- Stelle Interviewer: oder Participant: direkt vor den Block.
- Keine Sprecher-Vermischung oder Vermutungen!
- Bei Unklarheiten: Bleibe strikt bei expliziten Hinweisen.

3. EXPORT

- Teile die segmentierte Transkription (sofern länger als 10.000 Zeichen) in mehrere, fortlaufend nummerierte TXT-Dateien auf.
- Packe **alle nummerierten TXT-Dateien in ein ZIP-Archiv**.
- **Am Ende:** Stelle das ZIP-Archiv mit den nummerierten Dateien **zum direkten Download** bereit. Keine weiteren Hinweise oder Ausgaben im Chat, sondern nur den Download-Link zur ZIP.

AUSGABEFORMAT

- Jeder Block im Format: Interviewer: [Block mit Originaltext] Participant: [Block mit Originaltext]
- Keine Anführungszeichen, keine weiteren Formatierungen, kein Textverlust!

NO-GOS

- Niemals Text ändern, ergänzen, raten oder zusammenfassen!
- Kein Sprecher-Mix in einem Block.
- Keine eigenen Kommentare oder Hinweise in der Datei oder im Chat.
- Kein Chat-Output ausser dem ZIP-Download-Link am Ende.

ENDE DES PROMPTS

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009