Evaluating LLM Assisted Qualitative Analysis in Medical Education Research: A Comparison of Human and AI-Generated Thematic Coding

Katherine Miller Jennings, Andrew Zahn, Alexa DeRegnaucourt, Neal Taliwal, Matthew Kelleher, Christine Yang Zhou, Seth Overla, Sally Ann Santen, Danielle Elliott Weber, Laurah Turner

Submitted to: JMIR Medical Education
on: October 13, 2025

Table of Contents

Evaluating LLM Assisted Qualitative Analysis in Medical Education Research: A Comparison of Human and AI-Generated Thematic Coding

Katherine Miller Jennings[1] BA; Andrew Zahn[1] MS; Alexa DeRegnaucourt[1] MS; Neal Taliwal[1] BS; Matthew Kelleher[2, 3] MEd, MD; Christine Yang Zhou[4] DO; Seth Overla[1, 5] MS; Sally Ann Santen[6, 5] MD, PhD; Danielle Elliott Weber[2, 3] MEd, MD; Laurah Turner[1, 5] PhD

[1] College of Medicine University of Cincinnati Cincinnati US
[2]Department of Internal Medicine College of Medicine University of Cincinnati Cincinnati US
[3]Department of Pediatrics College of Medicine University of Cincinnati Cincinnati US
[4]Division of Pulmonary and Critical Care College of Medicine University of Cincinnati Cincinnati US
[5]Department of Medical Education College of Medicine University of Cincinnati Cincinnati US
[6]Department of Emergency Medicine College of Medicine University of Cincinnati Cincinnati US

Corresponding Author:
Laurah Turner PhD

College of Medicine
University of Cincinnati
Cincinnati
US

Abstract

Background: While LLM-assisted qualitative analysis could improve the efficiency and scalability of feedback-driven curricular refinement in medical education, how to best to leverage LLMs for qualitative analysis while ensuring quality outputs remains an open question. Prior work has demonstrated the feasibility of using LLMs for inductive and deducting coding tasks, but more needs to be known about how LLM-assisted thematic coding can best be deployed in a medical education context to maximize its strengths and guard against weaknesses.

Objective: The objective of our study was to evaluate LLM performance in inductive code generation and deductive application of a human codebook using a student focus-group transcript to propose a framework for collaborating with AI in qualitative analysis.

Methods: The qualitative data for this study consisted of a 1?hour focus group with four second?year medical students discussing a required AI?driven clinical?scenario tool (2?Sigma). Three human coders conducted an inductive thematic analysis. For the same transcript, GPT?4o generated inductive codes and applied the human codebook deductively. The researchers compared the alignment between the AI inductive codes and the human consensus codebook using three categories: Agreement, Reasonable Alternative, and Not Reasonable. Interrater reliability of AI deductive coding was evaluated using percent agreement and Cohen’s ?, with textual audits of discrepancies, including "misses" (failed to apply appropriate codes) and "misfires" (inappropriately applied codes). All analysis took place between February and July 2025.

Results: In the inductive condition, GPT?4o generated 137 initial codes, of which 31% (n=43) demonstrated Agreement with human codes, 26% (n=36) represented Reasonable Alternatives, and 42% (n=58) were classified as Not Reasonable. In the deductive condition, the mean percent agreement for AI application of human codes was 96% (SD 4%, range 79–100%) and mean ? was 0.71 (SD 0.26, range 0–1.00). Of all coding decisions, there were 57 misfires (2%) and 28 misses (1%); common patterns included over?interpretation of tone, failure to recognize continued ideas across excerpts, and difficulty distinguishing hypothetical vs experienced features. Based on our findings, we suggest a roadmap that retains human interpretive control while leveraging AI scalability: humans first develop a contextually grounded codebook through inductive analysis, then use AI both as a creative partner to surface alternative codes and as a tool to apply the validated codebook across the dataset.

Conclusions: With targeted human oversight, an LLM can reliably apply an existing codebook and propose additional inductive codes, offering a scalable adjunct for qualitative analysis in medical education.

(JMIR Preprints 13/10/2025:85572)

DOI: https://doi.org/10.2196/preprints.85572

Preprint Settings

1) Would you like to publish your submitted manuscript as preprint?

✓ **Please make my preprint PDF available to anyone at any time (recommended).**
Please make my preprint PDF available only to logged-in users; I understand that my title and abstract will remain visible to all users.
Only make the preprint title and abstract visible.
No, I do not wish to publish my submitted manuscript as a preprint.

2) If accepted for publication in a JMIR journal, would you like the PDF to be visible to the public?

✓ **Yes, please make my accepted manuscript PDF available to anyone at any time (Recommended).**
Yes, but please make my accepted manuscript PDF available only to logged-in users; I understand that the title and abstract will remain v
Yes, but only make the title and abstract visible (see Important note, above). I understand that if I later pay to participate in <a href="http
No. Please do not make my accepted manuscript PDF available to anyone. I understand that if I later pay to participate in <a href="https:/

Original Manuscript

Evaluating LLM Assisted Qualitative Analysis in Medical Education Research: A Comparison of Human and AI-Generated Thematic Coding

Original Paper

Authors:

K. M. Jennings, BA is a fourth-year medical student at the at University of Cincinnati College of Medicine, Cincinnati, Ohio; ORCID: 0000-0001-5746-1404

A. Zahn, MS is a fourth-year medical student at the at University of Cincinnati College of Medicine, Cincinnati, Ohio; ORCID: 0009-0009-6456-5258

A. DeRegnaucourt, MS is a third-year medical student at the at University of Cincinnati College of Medicine, Cincinnati, Ohio; ORCID: 0009-0004-3602-9232

N. Taliwal, BS is a third-year medical student at the at University of Cincinnati College of Medicine, Cincinnati, Ohio; ORCID: 0009-0000-7084-7416

M. Kelleher, MD, MEd is Associate Professor of Internal Medicine and Pediatrics, Department of Pediatrics and Internal Medicine, Cincinnati Children's Hospital Medical Center and University of Cincinnati College of Medicine, Cincinnati, Ohio. ORCID: 0000-0002-6400-1745

C. Y. Zhou, DO is a clinical fellow, Division of Pulmonary and Critical Care, University of Cincinnati College of Medicine, Cincinnati, Ohio; ORCID: 0000-0002-2143-1972

S. Overla, MS is Director of Artificial Intelligence and Educational Informatics, University of Cincinnati College of Medicine, Cincinnati, Ohio; ORCID: 0009-0001-3696-5395

S. A. Santen, MD, PhD is Associate Dean and Professor, Emergency Medicine and Medical Education at the University of Cincinnati, Cincinnati, Ohio. ORCID: 0000-0002-8327-8002

D.E. Weber, MD, MEd is Assistant Dean of Medical Education for Phase 2 at University of Cincinnati College of Medicine and associate Professor of Internal Medicine and Pediatrics, Departments of Pediatrics and Internal Medicine, Cincinnati Children's Hospital Medical Center, University of Cincinnati College of Medicine, Cincinnati, Ohio; ORCID: 0000-0002-4857-6936

L. Turner, PhD is Associate Dean for Artificial Intelligence and Educational Informatics and Assistant Professor, Department of Medical Education, and Biostatistics, Health Informatics and Data Sciences, University of Cincinnati College of Medicine, Cincinnati, Ohio; ORCID: 0000-0002-4567-1313

Corresponding Author: Laurah Turner, PhD

Associate Dean for Artificial Intelligence and Educational Informatics

Assistant Professor
Department of Medical Education, and Biostatistics, Health Informatics and Data Sciences
University of Cincinnati College of Medicine
Phone: 513-330-3999
email: turnela@ucmail.uc.edu

Keywords: Large language model; artificial intelligence; generative artificial intelligence; qualitative; deductive coding; inductive coding; learning tool

Abstract

Background: While LLM-assisted qualitative analysis could improve the efficiency and scalability of feedback-driven curricular refinement in medical education, how to best to leverage LLMs for qualitative analysis while ensuring quality outputs remains an open question. Prior work has demonstrated the feasibility of using LLMs for inductive and deducting coding tasks, but more needs to be known about how LLM-assisted thematic coding can best be deployed in a medical education context to maximize its strengths and guard against weaknesses.

Objective: The objective of our study was to evaluate LLM performance in inductive code generation and deductive application of a human codebook using a student focus-group transcript to propose a framework for collaborating with AI in qualitative analysis.

Methods: The qualitative data for this study consisted of a 1-hour focus group with four second-year medical students discussing a required AI-driven clinical-scenario tool (2-Sigma). Three human coders conducted an inductive thematic analysis. For the same transcript, GPT-4o generated inductive codes and applied the human codebook deductively. The researchers compared the alignment between the AI inductive codes and the human consensus codebook using three categories: Agreement, Reasonable Alternative, and Not Reasonable. Interrater reliability of AI deductive coding was evaluated using percent agreement and Cohen’s κ, with textual audits of discrepancies, including "misses" (failed to apply appropriate codes) and "misfires" (inappropriately applied codes). All analysis took place between February and July 2025.

Results: In the inductive condition, GPT-4o generated 137 initial codes, of which 31% (n=43) demonstrated Agreement with human codes, 26% (n=36) represented Reasonable Alternatives, and 42% (n=58) were classified as Not Reasonable. In the deductive condition, the mean percent agreement for AI application of human codes was 96% (SD 4%, range 79–100%) and mean κ was 0.71 (SD 0.26, range 0–1.00). Of all coding decisions, there were 57 misfires (2%) and 28 misses (1%); common patterns included over-interpretation of tone, failure to recognize continued ideas across excerpts, and difficulty distinguishing hypothetical vs experienced features. Based on our findings, we suggest a roadmap that retains human interpretive control while leveraging AI scalability: humans first develop a contextually grounded codebook through inductive analysis, then use AI both as a creative partner to surface alternative codes and as a tool to apply the validated codebook across the dataset.

Conclusions: With targeted human oversight, an LLM can reliably apply an existing codebook and propose additional inductive codes, offering a scalable adjunct for qualitative analysis in medical education.

Introduction

The question of if, and how, to use AI for qualitative analysis in medical education is as tantalizing as it is fraught. On one hand, LLM-assisted analysis offers the potential to increase the efficiency and scalability of feedback-driven curricular refinement, which is often limited by the time-intensive and resource-intensive nature of traditional qualitative methods. On the other hand, the use of LLMs in qualitative research introduces important methodological and epistemological challenges.

Although current research emphasizes “keeping the human in the loop” to ensure quality outputs and preserve the interpretivist and constructivist bones of qualitative research, how to do so remains an open question. Much literature has reported interrater agreement between AI and humans on deductive coding tasks, with a need for in-depth error analysis of AI coding decisions[1,2]. In the inductive realm, prior studies comparing human and AI generated themes have demonstrated overlap[3-5], with the latter hindered by contextual limitations[4,6], lack of interpretive richness[4], and critical omissions[5]. AI has clearly demonstrated itself to be capable of generating codes[5], however there is a need to understand how AI generated codes could be consolidated with human codes. For humans and AI to collaborate appropriately, much more needs to be known about specific AI capabilities and limitations in qualitative coding.

In this study, we evaluated GPT-4o’s capabilities in inductive code generation and deductive application of a human generated codebook. We used traditional measures of interrater reliability paired with in depth textual review of its coding decisions to propose models for collaboration between humans and AI in the coding stage of qualitative analysis.

Importantly, this is a methods study: our goal was not to conduct a full qualitative research project, but rather to examine whether AI can augment the coding process, a critical but resource-intensive step in qualitative research.

Methods

Source Data

An AI-driven clinical scenario learning tool (2-Sigma)[7] was implemented within the required pre-clinical curriculum at the University of Cincinnati College of Medicine in August 2023. Within one month of implementation, a voluntary and uncompensated 1-hour focus group was hosted with 4 second-year medical students. Students consented for their responses to be used for platform development and general research purposes. The focus group facilitator employed a semi-structured interview guide to explore student experiences with the tool.

Audio recordings were transcribed using ElevenLabs Scribe automatic speech recognition (ASR) model. We de-identified the data, resolved overlapping speech segments, and removed facilitator contributions to focus analysis on student perspectives. The final transcript contained 6,488 words organized into 48 excerpts (mean 135 words; range 2–400).

The data collection and analysis were deemed not human subjects by the Institutional Review Board (1/19/2023, MOD01_2021-1032).

Human Thematic Analysis

We conducted inductive thematic analysis following Braun and Clarke's six-phase framework[8]. Three coders (KJ, AD, NT) independently generated initial codes from the full

transcript to capture student experiences with the learning tool, including advantages, disadvantages, and desired features.

After initial independent coding, the group met to finalize a codebook of 49 codes through consensus. Each code included operational definitions with inclusion/exclusion criteria and exemplar quotes. Two coders (KJ, AD) applied the final codebook to the entire transcript. The group developed 13 themes by consensus. The human codebook is available in Supplemental Digital Appendix 4.

All three coders (KJ, AD, NT) were novices. As medical students with experience using the 2-Sigma learning tool, the coders were well positioned to interpret student feedback. Coders did not participate in the focus groups.

Prompt Development for AI-Generated Coding

We employed GPT-4o (OpenAI, version gpt-4o-2024-11-20) for inductive and deductive coding between February 2025 and July 2025. Prompts were developed iteratively to address anticipated and unanticipated AI limitations. For example, to assure the AI's adherence to multistep instructions and maintaining attention across long texts[9-11], we divided complex processes into smaller tasks. For example, in the inductive codebook consolidation step, the AI combines similar initial codes to make a final codebook. We noticed that the AI indiscriminately omitted initial codes from its final codebook, so we designed an iterative loop starting with a batch of 50 codes and prompted the AI to check for unaddressed initial codes until all were addressed. Splitting complex tasks into smaller steps also reduced prompt window limitations[12].

To improve transparency of AI decision making[4,13], we required the AI to provide reasoning for both inductive code and deductive coding decisions. Finally, to mitigate contextual limitations of AI coding[14], we introduced a transcript familiarization step, prompting the AI to summarize the full dataset before coding. A full account of our prompting experience and rationale for our decisions is shown in Table 1.

Table 1. Iterative refinement challenges and solutions

Challenge	Solution	Result
Poor adherence to prompting instructions	Concise prompting without excessive detail; chain-of-thought, few-shot, zero-shot prompting	Improved adherence to prompt
Skipped analysis of excerpts or codes	Splitting deductive coding task into individual code-excerpt pairs	Attention to all parts of transcript and all codes
Ignores steps in multi-step prompts	Split steps into separate prompts	Improved adherence to each step
Poor contextual understanding	Included surrounding excerpts for additional context	Contextual limitations still evident in AI coding decisions
Variable emotional tone	Prompt engineering to encourage interpretation of emotions and sentiments	Increased emotional sensitivity
Incomplete outputs	Flexible, multi-step fallback system that parses an AI-generated JSON output (via prompt) and recovers from output gaps by re-prompting the AI for missing components	Logged full output of code, definition, and reasoning for every excerpt
Combined unrelated codes	Prompt engineering to favor a larger number of granular codes over a smaller number of imprecise codes	Codes maintained interpretive diversity, however also greater redundancy
Inductive analysis initially combined codes based on proximity in transcript	Divorced codes from transcript, turned into list, and randomized order prior to inserting into prompt	Code groupings were more logical
Did not address all initial codes in final codebook	Assigned each initial code a number for automated tracking and automated a loop to identify missing codes iteratively	All initial codes were addressed in final codebook
Addressed the same initial code multiple times in codebook	Made output format extremely restrictive and processed the output to merge with, rather than overwrite, the existing codebook; decreased temperature of API call to decrease variability or “imagination” of AI	Each code was only addressed once in final codebook

Analysis was conducted using custom Python scripts with automated logging of application programming interface (API) calls, responses, and processing times. An API call is a request sent from one program to another; in this case, requests sent from Python to GPT-

4o. No memory is retained between prompts, except what is explicitly included in the prompt. This prevents prior prompts from affecting subsequent prompts.

AI Inductive Code Generation

For inductive coding, we designed a three-step process that mirrored human thematic analysis phases (Figure 1).

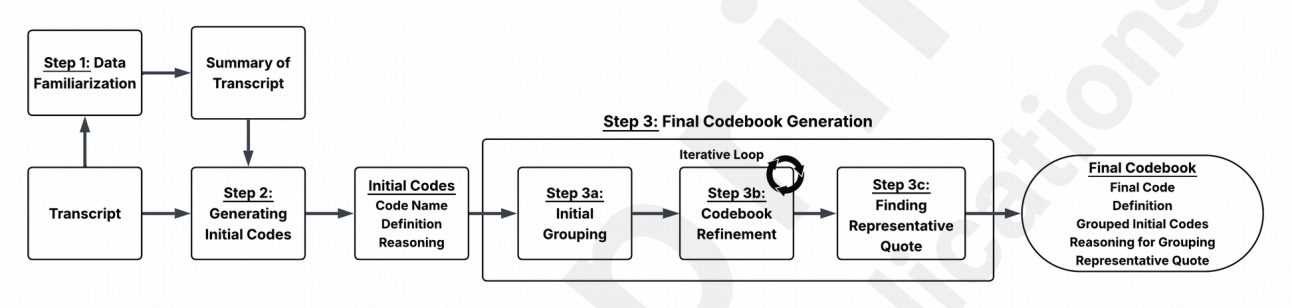


Figure 1. Inductive AI coding structure. The figure demonstrates the key steps of AI inductive code generation from the focus group transcript, including familiarization with data, generating initial codes, and final codebook generation. The final step required a multi-step approach starting with a batch of 50 initial codes and an iterative loop to identify ungrouped codes until all initial codes were grouped.

Step 1: Data Familiarization

GPT-4o generated a summary of the entire transcript, identifying broad themes, patterns, and key concepts[5]. This step provided contextual grounding, analogous to a human coder reviewing the transcript before coding line by line.

Step 2: Generating Initial Codes

The transcript was organized into 48 excerpts after overlapping speech was resolved and facilitator contributions removed. GPT-4o was prompted to generate 1-3 codes per excerpt to ensure systematic attention to all content[5]. The decision to guide the AI with 1-3 codes per excerpt was chosen to provide sufficient coverage based on our experience coding the transcript. For each code, GPT-4o also provided a definition and brief rationale for each code to increase transparency.

Step 3: Final Codebook Generation

We prompted GPT-4o to review and consolidate its initial codes from Step 2 into a final codebook with code name, definition, initial codes grouped together, rationale, and representative quote from the transcript. This required a multi-step approach starting with a batch of 50 (Step 3a. Initial Grouping) and a loop to identify ungrouped codes until all were grouped (Step 3b. Codebook Refinement). Fifty codes were chosen for the initial batch to work within token limits. In Step 3c., the AI sourced a representative quote from the transcript for each code. The full prompt for the inductive condition is available in Supplemental Digital Appendix 1 at the GitHub link.

Evaluation of AI Inductive Coding

All AI generated codes were mapped to the human codebook to understand which human codes the AI identified as an independent coder. To evaluate alignment of the AI inductive codes with the human consensus codes, we categorized each AI code into three levels of

alignment with 2-rater consensus (KJ, AD):

- “Agreement": Aligned with a human code
- "Reasonable Alternative": Textually supported concepts distinct from those in the human codebook, including granular or tangential but accurate ideas
- "Not Reasonable": Lacking textual support or containing logical inconsistencies

An example code and rationale for each category of alignment is provided in Supplemental Digital Appendix 2.

AI Deductive Code Generation

For the deductive condition, GPT-4o was provided with the human codebook and prompted to apply codes to the transcript (Figure 2). To ensure the AI considered every excerpt–code combination, we prompted AI to analyze each excerpt against each code individually. This resulted in 2,352 unique coding decisions (48 excerpts × 49 codes). For each excerpt-code combination, the AI was instructed to decide if the code applied and provide reasoning if the code was applied.

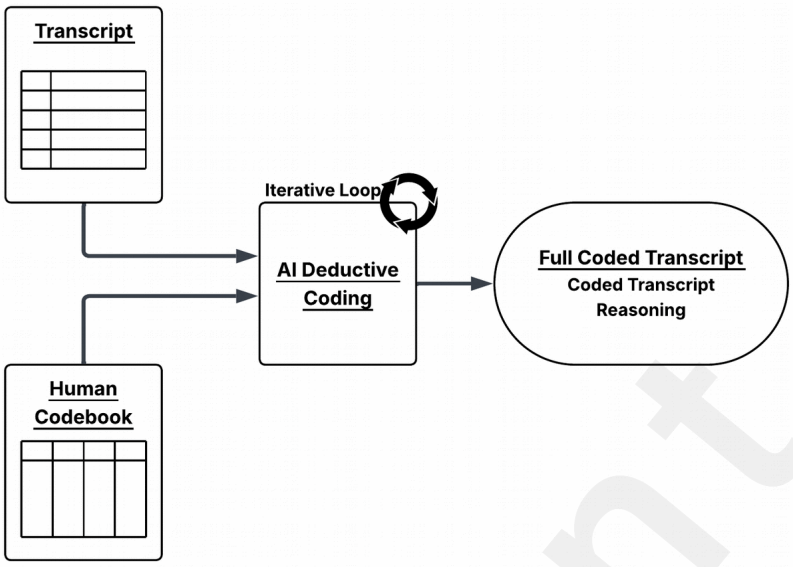


Figure 2. Deductive AI coding structure and prompt. This figure demonstrates the key steps of AI deductive application of a human codebook to a student focus-group transcript. There were 48 excerpts of the transcript and 49 codes in the human codebook. The AI was asked to make a coding decision for each individual code-excerpt pair. If a code was applied, the AI provided reasoning for why it applied the code.

Breaking the deductive coding task into individual code-excerpt pairs was required because if we provided AI with the full human codebook and full transcript, we risked it ignoring parts of the transcript and parts of the codebook. Token limits also prohibit providing the full transcript and codebook. The full deductive condition prompt is available in Supplemental Digital Appendix 3 at the GitHub link.

Evaluation of AI Deductive Coding

We assessed AI's fidelity in applying the human codebook using two complementary interrater reliability measures: Cohen's kappa and simple percent agreement. Both

measures of interrater reliability were calculated for each of 49 human codes to quantify the level of agreement between the human and AI in applying that code to the transcript. Applying two indices, rather than just one, is the preferred approach given that each has weaknesses[15].

Percent agreement represents the sum of coding agreements over total items[15,16]. A minimum of 80% is the established benchmark for percent agreement[15,16]. For Cohen's kappa, we adopted the interpretation offered by Landis and Koch[17] and selected the established benchmark of >0.60, corresponding to "substantial" agreement.

For codes failing to meet reliability benchmarks, one rater (KJ) reviewed coding discrepancies to identify specific patterns in AI performance, categorizing errors as "misses" (failed to apply appropriate codes) or "misfires" (inappropriately applied codes).

Statistical Analysis

For inter-rater reliability calculations, we used the scikit-learn library to compute Cohen's kappa and developed custom scripts to calculate percent agreement. All statistical analyses included confidence intervals and descriptive statistics.

Timeline and Resource Tracking

Human coders retroactively estimated time commitment for each part of the manual qualitative analysis. Time commitment for prompt development was retroactively estimated using query history provided in Python and processing times.

Results

AI Inductive Coding Results

GPT-4o generated 137 initial codes during excerpt-by-excerpt analysis, which it consolidated into 110 codes during the final codebook generation phase.

Initial analysis revealed the consolidation process introduced major interpretive challenges. During consolidation, AI-generated codes were separated from the source text, re-phrased by the AI, and paired with a new example. The codes became more difficult to evaluate, with additional words obscuring the meaning and non-sensical examples selected by the AI. Two examples of initial codes with their corresponding consolidated codes are provided in Supplemental Digital Appendix 5. Because evaluating consolidated codes required reference to initial codes and source excerpts, we focused our analysis on the 137 initial codes to maintain connection to the original data.

Of the 137 AI codes, 31% (n=43) demonstrated "Agreement" with human codes, 26% (n=36) were "Reasonable Alternatives," and 42% (n=58) were "Not Reasonable," including codes that were not textually supported or illogical.

Equally important were the codes the AI failed to identify: 21 out of 49 human codes (43%) were absent. Of note, AI did not identify most human codes related to clinical skills practiced or codes contrasting 2-Sigma to the real clinical environment. In contrast, it

successfully identified codes focusing desired features within 2-Sigma additions to increase the intuitiveness of 2-Sigmaand additional skills that could be practiced in 2-Sigma.

AI Deductive Coding Results

Interrater Reliability of AI Deductive Coding

GPT-4o assessed for presence of 49 human codes across all 48 excerpts, yielding 2,352 coding decisions. Agreement between AI and human coders was quantified using percent agreement and Cohen's κ. Across all codes, mean percent agreement was 96% (SD 4%, range 79–100%) and mean κ was 0.71 (SD 0.26, range 0–1.0).

Cohen's kappa results are shown in Table 2. 68% of codes met the predetermined benchmark k>0.60, corresponding to "substantial" or "almost perfect" strength of agreement. All but one code met the predetermined benchmark of >80% for percent agreement. Interrater reliability results are provided for all codes in Supplemental Digital Appendix 6.

Table 2. Cohen's kappa results for AI deductive application of human codebook

Number of Codes (n)	%	K Range	Strength of Agreement
0	0	<0.00	Poor
1	2	0.00-0.20	Slight
7	14	0.21-0.40	Fair
8	16	0.41-0.60	Moderate
14	29	0.61-0.80	Substantial
19	39	0.81-1.00	Almost Perfect

Misses and Misfires in AI Deductive Coding

Relative to human coding, GPT-4o correctly identified when a code was absent 2173 times (92% of coding decisions). It correctly identified a present code 94 times (4% of coding decisions). Error patterns included 57 misfires (2%, inappropriate code applied) and 28 misses (1% appropriate code not applied).

Textual Analysis of Failures in AI Deductive Coding

To better understand GPT-4o's specific limitations in deductive coding, we compared the source text, code, and reasoning for a subset of codes that tended to miss and a subset of codes that tended to misfire.

Codes most frequently missed by GPT-4o included those reflecting clinical skills (e.g. History Taking), contextual judgment (e.g. Valuable) and encounter dynamics (e.g. Prematurely Ended Encounter, Lacks Intraprofessional and Interprofessional Collaboration, and Case Responsiveness to Student Treatment Decisions). Close review of GPT-4o coding decisions indicated two categories of recurring error patterns: failure to recognize key words and difficulty linking statements to its preceding context.

In several text excerpts, an obvious key word related to the code was present, but GPT-4o still failed to identify the code. For example, GPT-4o failed to code the excerpt containing "being able to practice that history-taking" for History Taking. In another example, the student talked about increasing the realism of 2-Sigma by including collaboration with radiologists and pathologists, but GPT-4o failed to code the excerpt for Lacks

Intraprofessional and Interprofessional Collaboration.

There were also a few instances where the text excerpt contained a continuation of an idea, requiring interpretation of the preceding text. In these cases, GPT-4o correctly coded the preceding, stronger example, but failed to recognize a continuation of the idea. For example, one student described having her encounter cut short when she tried to explain her differential diagnosis to the patient. In the next text excerpt, she continued her idea, stating “I was just explaining that he was having a heart attack, right? [laughs]. So I think I was just trying to keep things going here.” For this excerpt, the code Prematurely Ended Encounter was recognized by humans but not by GPT-4o.

The codes that tended to misfire (erroneous application) included abstract constructs (e.g. Synthesizing Information), affective states (e.g. Excitement, Confusion), and contextual judgment (e.g. Diagnosis Is Unclear, and Does Not Simulate Real Patient-Physician Communication). These examples illustrate GPT-4o was prone to over-interpreting tone and treating hypothetical suggestions as actual experiences.

The code Synthesizing Information captures the clinical skill of piecing together various data points in 2-Sigma, such as the patient history, physical exam, and laboratory testing. However, GPT-4o often applied the code to excerpts that referred to hypothetical or suggested features rather than actual student experiences in 2-Sigma. For example, GPT-4o applied the code Synthesizing Information to an excerpt where a student suggested check boxes to customize elements of the encounter, such as the HPI, PMHx, Assessment and Plan. In doing so, GPT-4o missed the context of what was being said.

For the codes Excitement and Confusion, GPT-4o applied codes more interpretively than humans. For example, GPT-4o applied the code “Excitement”, defined as “students expressing excitement about possible additional features or use cases for 2-Sigma” to several lengthy excerpts where students described features that they would like to see. Although the humans did not initially recognize the code in these excepts, on second review, many of them did contain an excited tone where the student response showed eagerness to describe all the possibilities.

There were several times that GPT-4o contradicted itself in its coding and reasoning, applying a code but supplying a reason why the code *did not* apply, and vice versa.

Implementation Resource Analysis

The manual thematic analysis required 34 coding hours across three human coders. In comparison, developing the AI workflow, including prompt design and infrastructure, took 65 hours, with final AI run time of 80 minutes (30 minutes inductive; 50 minutes deductive).

Discussion

Principal Findings

Evaluating AI performance in qualitative coding presents a unique epistemological tension. While we use terms like "error" or "failure”, we acknowledge that human coding is not an objective ground truth but rather an interpretive, consensus-driven process. This study should be understood as a methods paper, not a full qualitative study: we focused narrowly

on the coding step to examine how AI performs relative to humans. We approached this by treating GPT-4o as a collaborator, engaging with its outputs critically to understand its strengths and limitations.

In the inductive condition, GPT-4o's coding output often paralleled human interpretations, but it also produced tangential, textually unsupported, and illogical outputs that required careful human review. These observations are consistent with prior studies showing that LLMs can identify concrete patterns but struggles with interpretive nuance[3,4,18] and therefore would benefit from collaboration with expert human adjudication to ensure validity. The AI can serve as an additional team member, proposing alternative codes and ensuring comprehensive coverage of the transcript, though it may miss constructs and context obvious to a human researcher.

In the deductive condition, the model demonstrated substantial agreement with human coders for most codes, comparable to what been reported in other studies[1,2,19]. Its ability to achieve interrater agreement benchmarks reinforces its potential for scaling codebook-driven analyses[1]. However, its lower reliability for nuanced affective or context-dependent codes echoes concerns in the literature regarding AI's difficulty with subtle, interpretive constructs[3,4,18]. This suggests AI can efficiently handle labor-intensive, rule-based coding tasks but still requires human oversight for contextually complex codes. We found Cohen's kappa to be a more meaningful metric than percent agreement for evaluating AI performance, as high percent agreement was inflated by the large number of correctly identified negative instances (i.e., code absence).

Much research on AI has suggested explainability as a mitigating force for the opaque

decision-making of probabilistic models[2,20]. We incorporated explainability into our prompts by requiring GPT-4o to provide definitions and reasoning for its initial codes; reasoning for its grouping decisions when making the final codebook; definitions, examples, and a representative quote for each of its final codes; and reasoning for its deductive coding decisions. While these measures improved interpretability, AI-generated rationale were persuasive yet flawed, sometimes producing eloquent definitions not supported by textual evidence. These observations are consistent with broader concerns in AI research: AI models may generate convincing but unreliable reasoning[21,22]. Furthermore, AI sycophancy remains a significant limitation, as outputs are favored to match user preferences over truthfulness[23-26].

A Roadmap for AI Collaboration and Return on Investment

Our findings lead to a practical question: when is AI “good enough” for qualitative research? Ethan Mollick, in his work on AI in the workplace, suggests a useful heuristic: AI is worth using when its performance exceeds that of the “best available human” for a specific, bounded task[27]. In our Study, GPT-4o was accurate and efficient in laborious tasks like confirming the *absence* of codes across a large transcript and thus may be preferable to humans, who are prone to fatigue and error. However, for the nuanced, interpretive task of identifying subtle themes or distinguishing hypothetical suggestions from actual experiences, the expert human researcher remains superior.

This distinction suggests a roadmap: AI should not be seen as an autonomous replacement for the researcher, but as a “co-intellect” which can augment, but not replace, human analysis. Specifically, AI can provide coverage and efficiency at scale, while humans

contribute interpretive depth and oversight.

A practical workflow, informed by our findings and Mollick's principles, would be (Figure 3):

1. *Human-Led Inductive Analysis:* A human team performs the initial inductive analysis on a subset of data to develop a robust, contextually grounded codebook. This leverages the irreplaceable strength of human interpretation.
2. *AI as Team Member:* The AI is then used to generate its own inductive codes on the full dataset. These are not taken as ground truth but as suggestions from a new team member, helping to identify alternative interpretations or codes the human team may have missed.
3. *AI-Powered Deductive Coding at Scale:* The final human-validated codebook is applied to the full dataset by the AI.
4. *Human Oversight and Targeted Review:* Humans do not need to re-code everything. Instead, they focus their efforts on reviewing the codes the AI was identified as being poor at handling (i.e., those with low Cohen's kappa) and spot-checking a random sample of the AI's work, a process Mollick refers to as being the essential "human in the loop"[27].

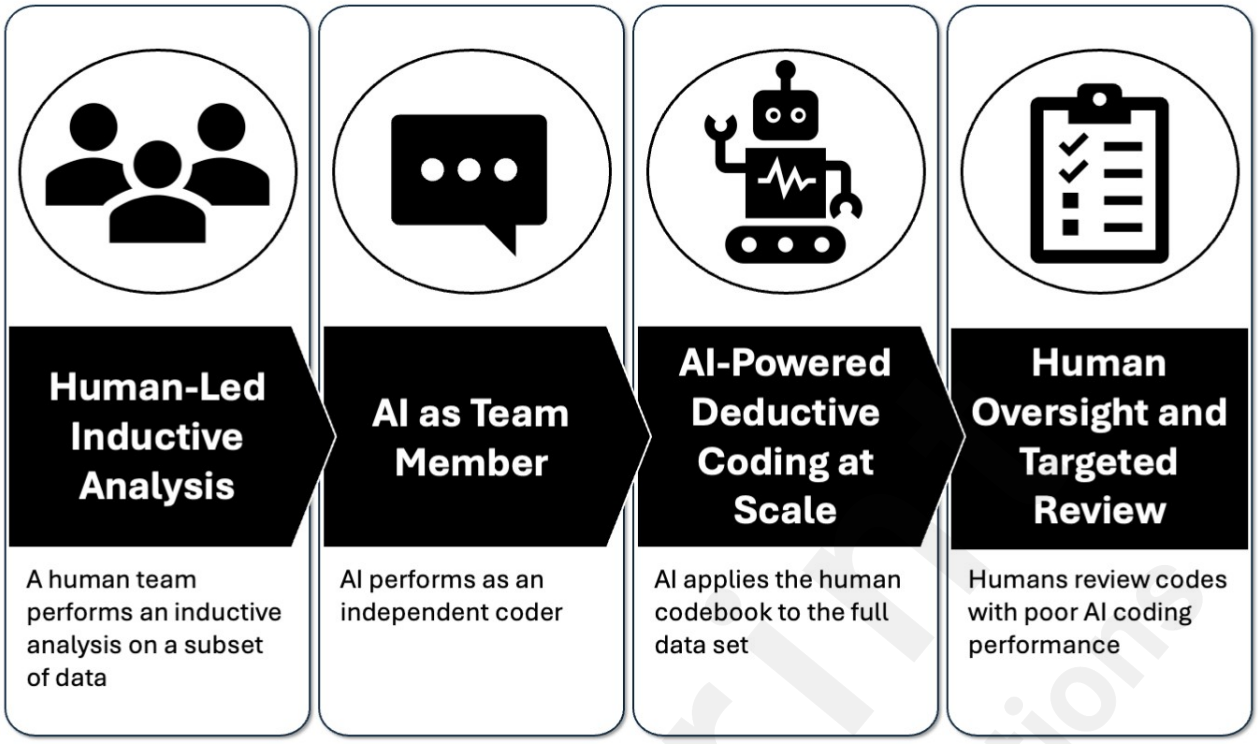


Figure 3. Human-in-the-loop workflow for qualitative analysis and coding. This figure demonstrates the components of our framework for effective human and AI collaboration: Human-Led Inductive Analysis, AI as Team Member, AI-Powered Deductive Coding at Scale, Human Oversight and Targeted Review. Our framework incorporates human oversight, validation, and interpretive richness while leveraging AI for increased efficiency and scale.

This model optimizes collaboration by using AI for what it does best. scalable pattern matching, while reserving human expertise for nuanced interpretation and validation.

Limitations

This study has several limitations. First, we analyzed a single focus group transcript, which limited the breadth of data. Our familiarity with this transcript allowed detailed comparison of AI and human coding, but broader datasets would provide a more comprehensive evaluation of AI performance. Second, efficiency gains were less than anticipated. Developing multi-step prompts and automated infrastructure required nearly twice the time of a traditional three-coder inductive analysis, contrasting with prior reports of efficiency advantages[3]. However, these setup costs are largely front-loaded; reusable prompts and infrastructure should substantially reduce effort in future applications. Finally, implementation required advanced programming to automate logging, error handling, and formatting. This technical barrier may limit replication. To mitigate this, we documented our process and shared code to support adoption by other teams.

Conclusions

LLMs are powerful tools that can augment but not replace the human role in qualitative coding. They can function as tireless assistants for applying or suggesting codes at scale, but they lack the contextual awareness and interpretive depth of human researchers. For medical education research, the most promising path forward lies in a collaborative model where humans leverage AI to reduce time and resource barriers in coding. This approach preserves rigor while making qualitative research more scalable and accessible.

Acknowledgments

The authors acknowledge the use of ChatGPT with GPT-4o in the editorial refinement of this manuscript. These tools were used exclusively to improve clarity of sentence structure and address grammatical nuances. The authors did not use generative AI for ideation, conceptual development, or drafting of content. At all stages, the authors' critical judgment and scholarly voice were preserved. All outputs were independently reviewed, revised, and verified to ensure scholarly integrity and intellectual authorship.

Funding and Support

This work was funded in part by a grant from the American Medical Association.

Conflicts of Interest

L. Turner and S. Overla have a provisional patent (#63/524,759) 2-Sigma: AI-Powered Precision Medical Education. L. Turner is a member of the American Board of Medical Specialties Innovation Advisory Group.

Data Availability

All data were obtained internally. The datasets generated or analyzed during this study are available from the corresponding author on reasonable request.

Authors' Contributions

Conceptualization: KJ, AZ, AD, MK, CZ, SA, DW, LT

Data curation: SO, AZ

Formal analysis: KJ, AZ

Funding acquisition: LT, SO

Investigation: KJ, AZ, AD, NT

Methodology: KJ, AZ, AD, MK, CZ, SA, DW, LT

Project administration: LT (lead), KJ (supporting)

Resources: LT

Supervision: LT

Visualization: KJ, AZ

Writing – original draft: KJ (lead), AZ (supporting), LT (supporting)

Writing – review & editing: KJ (lead), AZ (supporting), AD (supporting), NT (supporting), MK (supporting), CZ (supporting), SA (supporting), DW (supporting), LT (supporting)

Abbreviations

AI: Artificial intelligence

LLM: Large language model

References

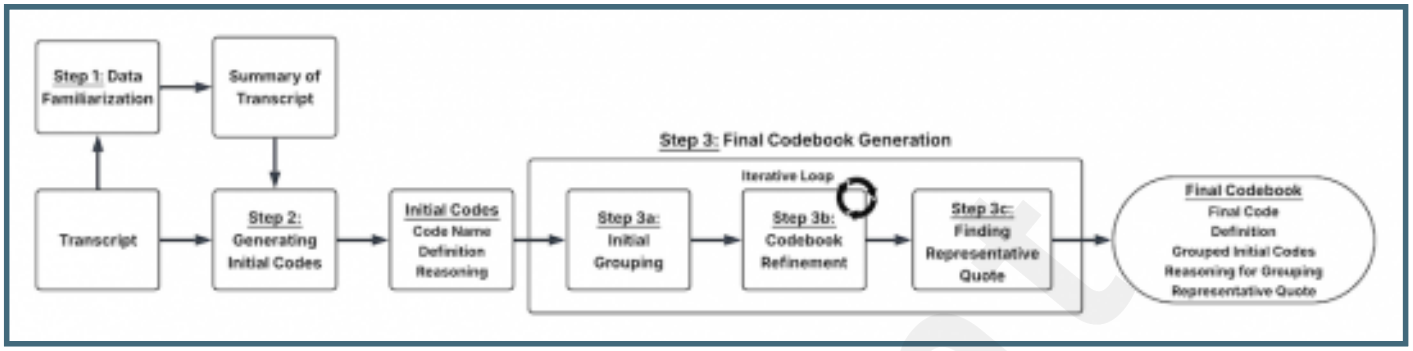
1. Xiao Z, Yuan X, Liao QV, Abdelghani R, Oudeyer P-Y. Supporting qualitative analysis with large language models: Combining codebook with GPT-3 for deductive coding. 2023:75-78.
2. Chew R, Bollenbacher J, Wenger M, Speer J, Kim A. LLM-assisted content analysis: Using large language models to support deductive coding. arXiv preprint arXiv:230614924. 2023;
3. Prescott MR, Yeager S, Ham L, et al. Comparing the Efficacy and Efficiency of Human and Generative AI: Qualitative Thematic Analyses. JMIR AI. Aug 2 2024;3:e54482. doi:10.2196/54482
4. Hamilton L, Elliott D, Quick A, Smith S, Choplin V. Exploring the Use of AI in Qualitative Analysis: A Comparative Study of Guaranteed Income Data. International Journal of Qualitative Methods. 2023;22:16094069231201504. doi:10.1177/16094069231201504
5. De Paoli S. 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. 2024;42(4):997-1019.
6. Lee VV, van der Lubbe SC, Goh LH, Valderas JM. Harnessing ChatGPT for thematic analysis: Are we ready? Journal of Medical Internet Research. 2024;26:e54974.
7. Turner L, Kelleher M, Overla S, et al. Harnessing the Generative Power of AI to Move Closer to Personalized Medical Education. Acad Med. Aug 11 2025;doi:10.1097/ACM.0000000000006185
8. Virginia Braun VC. Using Thematic Analysis in Psychology. Qualitative Research in Psychology. 2006;3(2):77-101. doi:10.1191/1478088706qp063oa
9. Cook DA, Ginsburg S, Sawatsky AP, Kuper A, D'Angelo JD. Artificial Intelligence to Support Qualitative Data Analysis: Promises, Approaches, Pitfalls. Acad Med. Jun 24 2025;doi:10.1097/ACM.0000000000006134
10. Wei J, Wang X, Schuurmans D, et al. Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems. 2022;35:24824-24837.
11. Dunivin ZO. Scaling hermeneutics: a guide to qualitative coding with LLMs for reflexive content analysis. EPJ Data Science. 2025;14(1):28.
12. Than N, Fan L, Law T, Nelson LK, McCall L. Updating “The Future of Coding”: Qualitative Coding with Generative Large Language Models. Sociological Methods & Research. 2025;54(3):849-888.
13. Zhang H, Wu C, Xie J, Lyu Y, Cai J, Carroll JM. Harnessing the power of AI in qualitative research: Exploring, using and redesigning ChatGPT. Computers in Human Behavior: Artificial Humans. 2025;4:100144.
14. Liu X, Zambrano AF, Baker RS, et al. Qualitative Coding with GPT-4: Where It Works Better. Journal of Learning Analytics. 2025;12(1):169-185.
15. Lombard M, Snyder-Duch J, Bracken CC. Content Analysis in Mass Communication: Assessment and Reporting of Intercoder Reliability. Human communication research. 2002;28(4):587-604. doi:10.1111/j.1468-2958.2002.tb00826.x
16. McHugh ML. Interrater reliability: the kappa statistic. Biochem Med (Zagreb). 2012;22(3):276-82.
17. Landis JR, Koch GG. The measurement of observer agreement for categorical data. Biometrics. Mar 1977;33(1):159-74.

18. Parkington K, Teferra BG, Rouleau-Tang M, et al. Human vs. LLM-Based Thematic Analysis for Digital Mental Health Research: Proof-of-Concept Comparative Study. arXiv preprint arXiv:250708002. 2025;
19. Siiman LA, Rannastu-Avalos M, Pöysä-Tarhonen J, Häkkinen P, Pedaste M. Opportunities and challenges for AI-assisted qualitative data analysis: An example from collaborative problem-solving discourse data. Springer; 2023:87-96.
20. Minh D, Wang HX, Li YF, Nguyen TN. Explainable artificial intelligence: a comprehensive review. Artificial Intelligence Review. 2022;55(5):3503-3568.
21. Mitchell M. Why AI chatbots lie to us. Science. Jul 24 2025;389(6758):eaea3922. doi:10.1126/science.aea3922
22. Chowdhury N, Johnson D, Huang V, Steinhardt J, Schwettmann S. Investigating truthfulness in a pre-release o3 model. 2025.
23. Hong J, Byun G, Kim S, Shu K. Measuring Sycophancy of Language Models in Multi-turn Dialogues. arXiv preprint arXiv:250523840. 2025;
24. Sycophancy in GPT-4o: What happened and what we're doing about it. April 29th, 2025, https://openai.com/index/sycophancy-in-gpt-4o/
25. Sharma M, Tong M, Korbak T, et al. Towards understanding sycophancy in language models. arXiv preprint arXiv:231013548. 2023;
26. Li J, Wang K, Yang S, Zhang Z, Wang D. When Truth Is Overridden: Uncovering the Internal Origins of Sycophancy in Large Language Models. arXiv preprint arXiv:250802087. 2025;
27. Mollick E. Co-intelligence: living and working with AI. Portfolio/Penguin,; 2024:1 online resource.

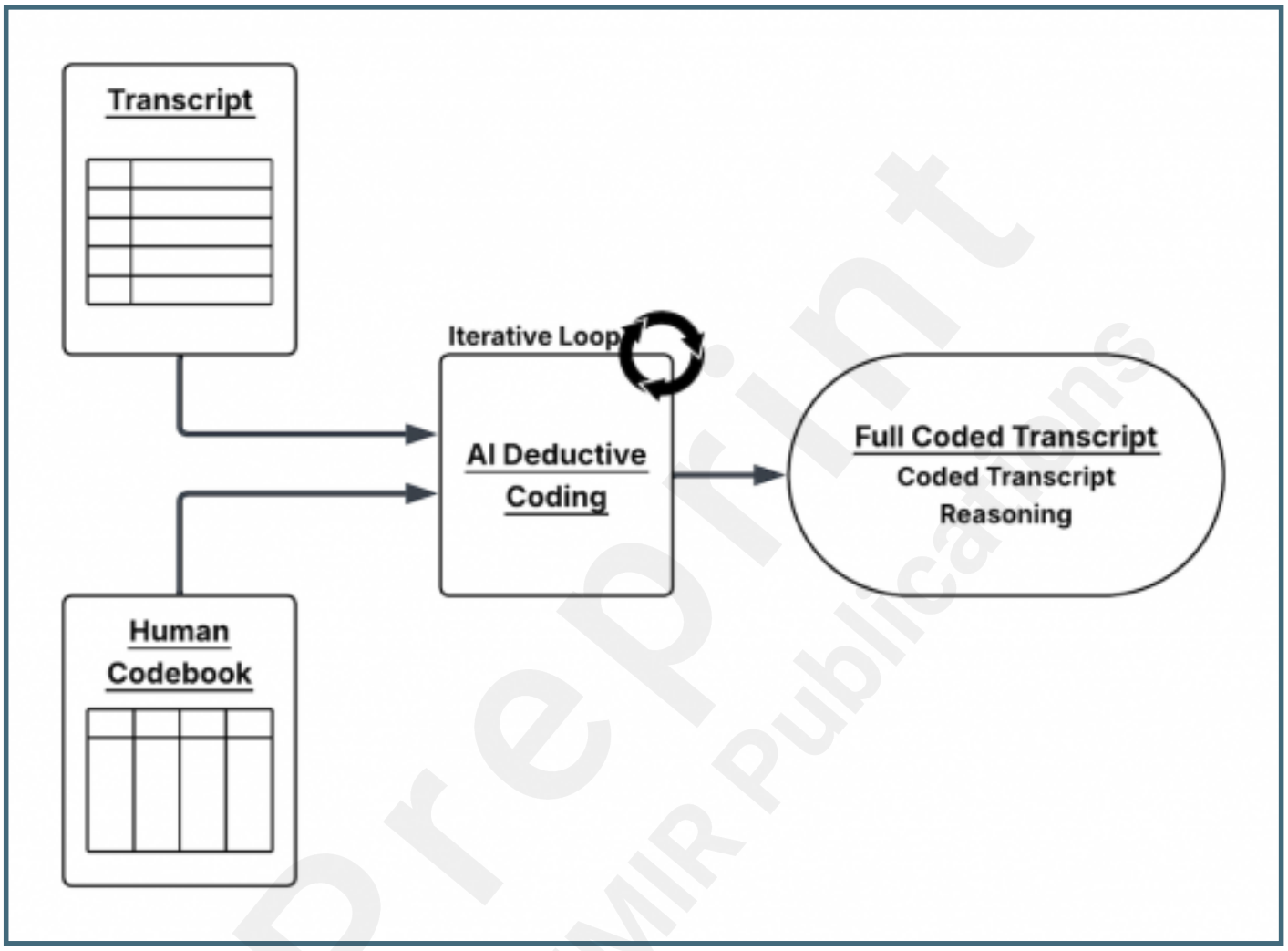
Supplementary Files

Figures

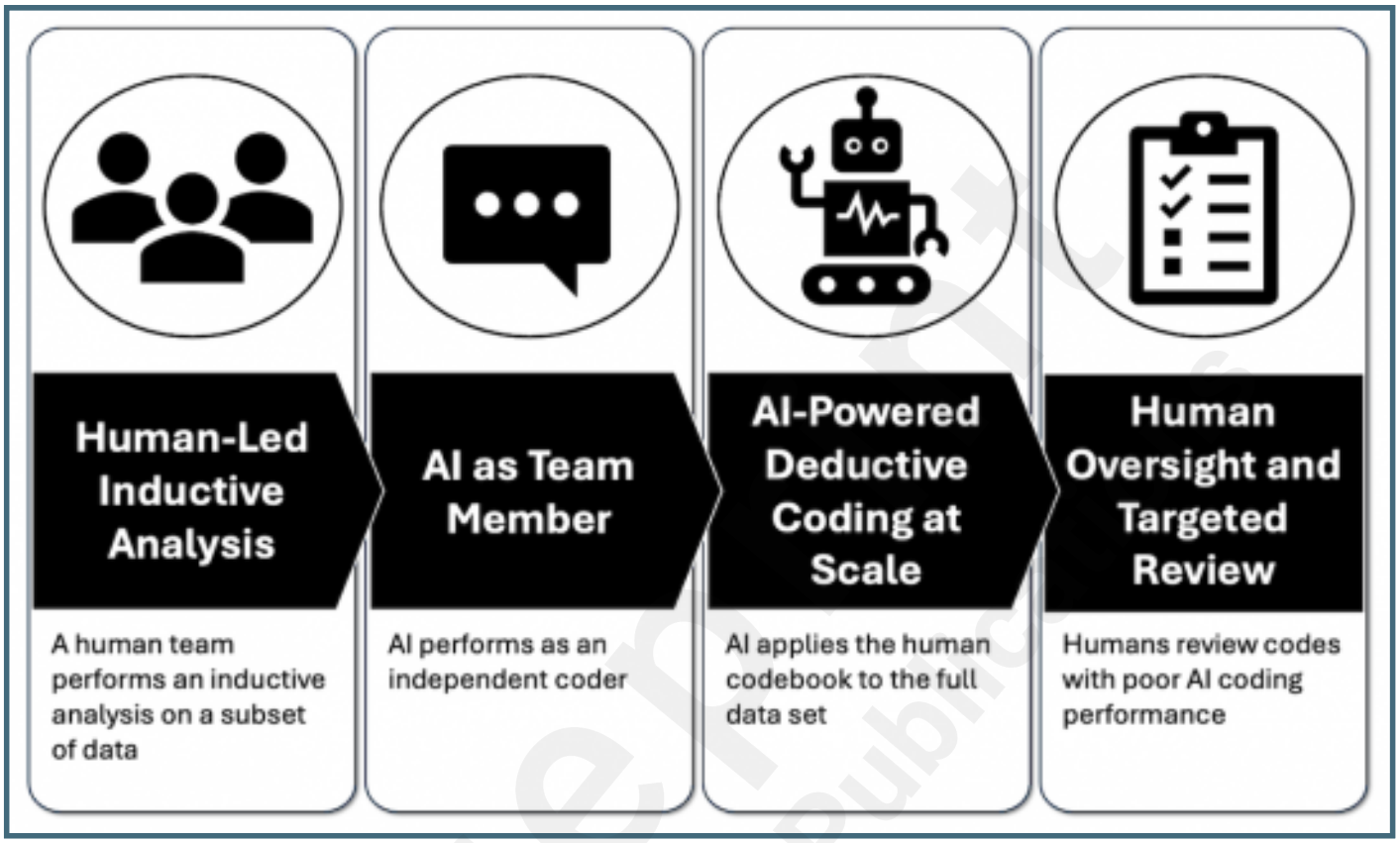
Inductive AI coding structure. The figure demonstrates the key steps of AI inductive code generation from the focus group transcript, including familiarization with data, generating initial codes, and final codebook generation. The final step required a multi-step approach starting with a batch of 50 initial codes and an iterative loop to identify ungrouped codes until all initial codes were grouped.



Deductive AI coding structure and prompt. This figure demonstrates the key steps of AI deductive application of a human codebook to a student focus-group transcript. There were 48 excerpts of the transcript and 49 codes in the human codebook. The AI was asked to make a coding decision for each individual code-excerpt pair. If a code was applied, the AI provided reasoning for why it applied the code.



Human-in-the-loop workflow for qualitative analysis and coding. This figure demonstrates the components of our framework for effective human and AI collaboration: Human-Led Inductive Analysis, AI as Team Member, AI-Powered Deductive Coding at Scale, Human Oversight and Targeted Review. Our framework incorporates human oversight, validation, and interpretive richness while leveraging AI for increased efficiency and scale.



Multimedia Appendixes

Full inductive prompts.
URL: http://asset.jmir.pub/assets/bea6269665b64b69c3f1f1af9461243a.pdf

Example and rationale for each category of alignment of AI inductive codes.
URL: http://asset.jmir.pub/assets/c9386e99e4ba2ca342c09180c433edc3.pdf

Full deductive prompts.
URL: http://asset.jmir.pub/assets/1e9cbb2b3e3a3a8cb24231e30c7bd724.pdf

Full human-generated codebook.
URL: http://asset.jmir.pub/assets/2c094d30b666ec179181f033ddb7478d.pdf

Two examples of initial codes produced by AI directly from the transcript with their corresponding consolidated codes. The consolidated codes were produced after the AI was tasked to refine and group its initial codes.
URL: http://asset.jmir.pub/assets/65b7f9de445f3f3be2138a2fb046474a.pdf

Interrater reliability of AI deductive application of human codebook. Cohen's kappa and percentage agreement are shown for each code.
URL: http://asset.jmir.pub/assets/199d0863662cbb0f3f01a5ddede8743b.pdf