Investigating the use of LLMs in Deductive Coding within the Context of Software Engineering

Samuel Mercês
State University of Feira de Santana
Feira de Santana, Brazil
samuelmerces@outlook.com

Moaath Alshaikh
Federal University of Bahia
Salvador, Brazil
moaathalshaikh@ufba.br

Gabriel Moraes
State University of Feira de Santana
Feira de Santana, Brazil
gcmorais66@gmail.com

Lucca de A. H. Coutinho
State University of Feira de Santana
Feira de Santana, Brazil
lucca.ahc@gmail.com

Glauco Carneiro
Federal University of Sergipe
São Cristôvão, Brazil
glauco.carneiro@dcomp.ufs.br

Manoel Mendonça
Federal University of Bahia
Salvador, Brazil
manoel.mendonca@ufba.br

José Amancio Macedo Santos
State University of Feira de Santana
Feira de Santana, Brazil
zeamancio@uefs.br

Abstract

Context: The growing emphasis on human and social factors in Software Engineering (SE) has increased the demand for qualitative studies. However, researchers face the challenge of balancing interpretive depth with the scalability required by modern software datasets. While Large Language Models (LLMs) emerge as promising tools for qualitative analysis, empirical evidence regarding their methodological rigor in subjective SE tasks remains scarce. **Aim**: This paper evaluates the effectiveness and feasibility of LLMs for deductive coding within SE qualitative workflows, focusing on consistency and alignment with human judgment. **Method**: We conducted a comparative study using the Gemini model to analyze SE data sources across varying levels of complexity. We evaluated how prompting strategies (One-shot vs. Few-shot) influence inference consistency. We also performed an interpretive taxonomical analysis to identify systematic patterns in AI-generated classifications compared to a human-validated baseline. **Results**: Our findings indicate a *conditioned feasibility*. The model demonstrated high competence in exhaustive screening but exhibited an "Excessive Granularity Bias", frequently overlooking subtle subjective nuances in favor of literal interpretations. It is worth noting that the One-shot strategy emerged as the only transversally viable approach across all dimensions of the study. Conversely, the Few-shot strategy faced operational limitations in two dimensions due to architectural constraints associated with multi-label classification. Furthermore, within the dimensions where both strategies were compared, the inclusion of contextual examples proved inefficient, ultimately increasing the rate of Superior Human Coding. **Conclusion**: This study systematizes findings correlated with data density and identifies specific SE use cases for LLM integration. We contribute by repositioning the researcher's role from a manual coder to an inference curator, providing a path toward more scalable yet rigorous qualitative research in SE.

Keywords

LLM, Qualitative Research, Deductive Coding, Prompt Engineering

1 Introduction

In recent decades, qualitative studies have gained prominence in technically dominated fields, as modern phenomena of interest are increasingly sociotechnical [21]. This trend is particularly evident in SE, a discipline that has historically concentrated efforts on technical aspects aimed at building functional and reliable systems [27, 33, 36]. However, the growing complexity of projects has highlighted the importance of non-technical skills, demonstrating that project success depends on the balance between human and technological factors [21, 33, 34].

There is growing recognition of the value of empirical qualitative studies in SE for understanding human and social factors that affect team interactions and software quality [10, 21, 34, 36]. However, qualitative analysis—particularly textual coding—is time-consuming and demands methodological rigor. It is also vulnerable to researcher subjectivity and bias, which can compromise validity and produce inconsistencies when handling large, interpretively rich datasets [3, 14, 20, 38, 42].

Given this scenario, LLMs have emerged as promising instruments to support qualitative research [3, 31, 38]. These models facilitate task automation, such as coding and result synthesis [3, 8]. Their ability to quickly identify patterns and understand context enables the revelation of findings that might otherwise go unnoticed in human analysis [3, 18, 42]. Furthermore, their ability to process natural language promotes a new paradigm of human-AI collaboration [29, 42].

While promising, the use of LLMs in qualitative research poses critical challenges [14, 20, 38, 42]. Although studies indicate satisfactory agreement rates between algorithmic and human coding [6, 8, 38, 42], doubts persist regarding these models' ability to capture interpretive subtleties and handle the contextual sensitivity inherent in qualitative data [3, 17, 31]. Additionally, the occurrence of "hallucinations" and response variability based on prompt formulation and model versions can compromise result reproducibility [3, 11, 14, 22].

Furthermore, the literature reiterates the need for meticulous evaluations of these tools' limitations [14, 32, 38]. Such an evaluation is complex due to the multiplicity of interdependent factors influencing their application, such as data characteristics, model architecture, prompting strategies, and fine-tuning techniques. From this perspective, knowledge regarding the characterization of favorable scenarios for using LLMs in qualitative studies is still nascent, with lingering doubts about when and how these models offer effective support. This gap demands systematic investigations to understand their strengths and limits under different conditions of use, contributing to a responsible and rigorous adoption of the technology.

Despite growing interest in applying generative AI to qualitative analysis, the empirical literature in SE remains limited. Early studies assessed the basic capabilities of language models for deductive coding in generic domains [38, 42], or focused on inductive thematic categorization [1, 8, 14]. However, to the best of our knowledge, there are no empirical evaluations that systematically examine the use of LLMs on deductive frameworks applied to software-engineering-specific data.

To address this methodological gap, the primary objective of this study is to evaluate the technical and methodological feasibility of using LLMs for the qualitative analysis of sociotechnical software-engineering data. This work focuses specifically on deductive coding—a systematic qualitative method in which a pre-defined theoretical framework or codebook is strictly applied to categorize raw textual data [13]. By transitioning the analysis from abstract conceptual definitions to concrete automated verification, this investigation establishes a baseline for incorporating AI into qualitative workflows.

In practice, we conducted a comparative study using the Gemini model against a peer-validated human baseline. We analyzed how different prompting strategies (One-shot vs. Few-shot) influence inference consistency. We also performed an interpretive analysis to identify systematic patterns between AI and human classifications, resulting in a taxonomy representing patterns of agreement and divergence between coders. Based on this taxonomy, potential use cases for integrating LLMs into the qualitative research workflow were identified. Consequently, this work advances understanding of how to incorporate AI into qualitative methodologies while preserving the rigor and contextual sensitivity inherent to the field.

The remainder of this paper is organized as follows: Section 2 provides the background for using LLMs in qualitative research. Section 3 details the experimental protocol. Section 4 presents the empirical results through the lens of the divergence taxonomy. Section 5 discusses the model's implications and feasibility. Section 6 addresses threats to validity, and Section 7 presents the conclusions.

2 Background

This section contextualizes the rise of sociotechnical qualitative studies in SE and the operational challenges of their analysis. It explores the feasibility of LLMs as tools for assisted automation, and subsequently discusses Prompt Engineering, the challenges and limitations of using LLMs in qualitative research, and related work.

2.1 Qualitative Research in Software Engineering

Qualitative research is a methodological approach focused on exploring and interpreting complex social phenomena, investigating experiences, perceptions, and meanings attributed by individuals [13]. It is characterized by reasoning anchored in human understanding and contextual immersion, allowing for the identification of the diversity and intrinsic characteristics of a phenomenon [37]. The investigative process is marked by data collection and an interpretive analysis that moves from specific instances to the construction of general themes [13].

The growth of qualitative studies in SE reinforces the discipline's sociotechnical view, where development success depends not only on technical metrics but on the interplay between human and technological factors [19, 21, 36]. Research into constructs such as empathy, psychological safety, emotional intelligence, and team dynamics underscores the need for methods that capture social nuances with the depth provided by qualitative and mixed approaches [10, 15, 28, 34]. At the same time, analyzing unstructured textual data remains labor-intensive and prone to subjectivity, creating practical and validity challenges for manual coding in large, interpretively rich datasets [3, 8, 12, 13].

To mitigate these risks and ensure rigor, the literature prescribes protocols such as investigator triangulation and thick description of the context [5, 44]. Although fundamental to research credibility, such strategies add layers of operational complexity, making the analytical process burdensome in projects with large data volumes. This scenario justifies the search for emerging technologies that assist in systematizing analytical work without sacrificing interpretive sensitivity.

The core of qualitative analysis lies in coding — the technical process of labeling, organizing, and categorizing textual data to identify underlying patterns and themes [12, 13]. Our work focuses specifically on deductive coding (or theory-driven coding), in which the researcher uses a pre-existing codebook to map the data [14, 42]. Unlike the inductive approach, which seeks the emergence of themes organically, the deductive method requires rigorous alignment between raw data and pre-established conceptual definitions. This task, while systematic, becomes burdensome and susceptible to cognitive fatigue when executed manually at scale, justifying the investigation of computational support for its optimization [4, 8].

2.2 LLMs in Qualitative Research

The emergence of LLMs, such as ChatGPT, LLaMA, and Gemini, has introduced a new paradigm in textual data analysis [20]. Based on the *Transformer* architecture, these models utilize self-attention mechanisms to process complex contextual dependencies, enabling a semantic understanding that transcends the simple keyword counting characteristic of traditional Natural Language Processing (NLP) [32, 39]. In qualitative research, this capacity translates into the ability to identify themes, summarize narratives, and perform coding tasks [3, 20, 31, 38, 42].

While promising, the operational effectiveness of LLMs as qualitative copilots depends on hyperparameter optimization. Parameters such as temperature regulate response randomness—where

values close to zero favor determinism—while the context window delimits the volume of information the model retains simultaneously [22, 42]. Effective prompt engineering acts as the primary programming interface to guide these probabilistic behaviors through In-context Learning, ensuring analytical precision in high-dimensional qualitative tasks [2, 29].

2.3 Prompt Engineering

Prompt Engineering is the activity of formulating instructions and examples that guide LLM behavior, serving as a critical component to ensure precision in tasks with high semantic specificity [2, 29, 42]. Given the models' sensitivity to input formulation, effective prompt engineering allows for bias mitigation and inference control through In-context Learning—a technique where the model adjusts its responses based on examples provided within the prompt itself, without needing to modify its internal parameters [23, 29]. The complexity of prompt engineering is such that the emergence of "human-level prompt engineers" is already being discussed [22].

Prompt strategies vary according to the density of contextual information provided; the following are highlighted in the literature:

- **Zero-shot:** This strategy consists of presenting direct instructions about the task without providing reference examples. While it simplifies artifact construction, its effectiveness in qualitative analysis may be limited by the lack of interpretive benchmarks for the model [7, 42].
- **One-shot and Few-shot:** These strategies enhance instructions by including, respectively, one or multiple input-output examples. These techniques demonstrate the expected format, tone, and rigor to the model, and are fundamental for stabilizing classification and deductive coding tasks [2, 29, 42].
- **Chain-of-Thought (CoT):** A strategy that encourages the model to break down reasoning into logical steps before presenting the final answer. In addition to boosting accuracy in complex problems, CoT offers greater interpretability by making the model's inference process explicit [2, 29].

Beyond these categories, the literature describes complementary approaches such as persona definition—which guides the LLM to adopt a specific role (e.g., a senior researcher)—and iterative optimization to continuously refine instructions, reduce ambiguities, and improve alignment with research objectives [25, 45]. Although Zero-shot and Chain-of-Thought prompting are prominent paradigms in prompt engineering, we deliberately excluded them from our experimental design: zero-shot lacks the structural and formatting constraints needed to reproduce qualitative outputs, and CoT adds multi-step conversational overhead that can distort the localized semantic accuracy required for immediate codebook applications.

2.4 Challenges and Limitations of LLMs in Qualitative Research

Despite their analytical potential, integrating LLMs into qualitative research raises critical validity challenges. The most prominent is the occurrence of hallucinations, where the model generates factually incorrect inferences—or those disconnected from the raw data—with a high degree of confidence [8, 38]. This propensity for "non-factual creativity" is intrinsic to the probabilistic nature of the models; even with parameter adjustments to favor determinism (e.g., temperature close to zero), the integrity of the results remains dependent on verification [14, 16].

Another challenge lies in algorithmic biases and contextual sensitivity. LLMs trained on large internet corpora may replicate social and cultural prejudices, which becomes particularly problematic in analyses seeking to capture nuances of human experience [20, 32]. Moreover, slight variations in prompt formulation can lead to divergent outputs, affecting the reproducibility and reliability of the analysis [8, 14]. This lack of contextual reasoning—often described as an "epistemological incongruity" due to the AI's lack of lived experience—can result in simplifications that ignore ethical or cultural subtleties present in qualitative data [17, 31].

Given these limitations, the literature converges toward the Human-in-the-loop (HITL) paradigm [40]. In this model, AI does not act as an autonomous replacement but as an assistant under human supervision. The researcher assumes the role of an Inference Curator, responsible for validating the coherence of suggested categories and filtering hallucinations [3, 43]. This active curation is the fundamental mechanism to ensure that computational efficiency does not compromise the interpretive depth and epistemic integrity of qualitative research [31].

2.5 Related Work

This section positions the present research within the literature on the application of LLMs in qualitative analysis. The rise of these tools has driven comparative studies evaluating their efficacy against human coding, aiming to understand their capabilities, limitations, and the potential for human-AI collaboration [1, 6, 38, 43].

2.5.1 Deductive Coding Studies.

The use of LLMs in deductive coding has been explored under different rigor metrics. Xiao et al. [42] utilized ChatGPT 3.0 to analyze children's curiosity, achieving substantial agreement (Cohen's Kappa of 0.61) and demonstrating that a codebook-centered One-shot strategy generates significant performance gains. In a complementary study, Tai et al. [38] proposed the LLMq quotient, using 160 iterations with ChatGPT 3.5 to stabilize coding reliability. The results showed consistency exceeding 84%, in some cases surpassing human reliability; nonetheless, the authors reinforce the need for continuous supervision due to the risk of hallucinations.

2.5.2 Hybrid and Comparative Studies.

Recent investigations contrast assisted automation with manual analysis. Bijker et al. [8] explored ChatGPT for automating data annotation under the Theoretical Domains Framework (TDF), obtaining a Kappa between 0.56 and 0.73. By contrast, Al-Fattal and Singh [1] conducted a collaborative autoethnography, concluding that while AI offers efficiency and scalability, it lacks the interpretive depth and cultural context present in manual analysis. In the field of inductive thematic analysis, De Paoli [14] observed that although models like ChatGPT 3.5-Turbo can infer main themes, they operate primarily on the formal structure of language rather than on deep human meaning.

This limitation is discussed by Messner et al. [31], who introduce the concept of "epistemological incongruities", suggesting that AI,

being "devoid of experience", requires the researcher to act as an inducer of "analytical persuasion" via strategic prompts. Meanwhile, Bano et al. [3] recognize the potential of LLMs to identify themes and detail analyses, but warn against over-reliance, given the frequent lack of consensus between AI insights and those of human specialists.

Unlike purely positivist approaches that seek total automation via binary accuracy metrics, the present research adopts a constructivist and interpretivist perspective. Rather than treating the divergence between AI and humans merely as statistical error, we investigate these discrepancies to identify patterns of agreement and divergence between classifications. In doing so, we position the LLM not merely as a classifier, but as a dialogic agent capable of challenging predefined categories and revealing semantic nuances that might otherwise go unnoticed in human analysis.

3 Methodology

The experimental protocol of this investigation was designed to evaluate the technical and methodological feasibility of using LLMs for the deductive coding of qualitative data. Since the reuse of peer-validated, inductively derived codebooks for subsequent automated auditing is an emerging scenario in SE, this study demands a systematic approach. Consequently, we ensure a formalized and reproducible empirical design by adapting the five foundational stages prescribed by Wohlin et al. [41]: scoping, planning, execution, analysis and interpretation, and reporting.

Concretely, this means treating a consolidated qualitative artifact as a stable local theory, which transforms the original inductive outputs into an objective benchmark for deductive classification. Building on this, we adopt a mixed-methods approach that treats anomalies not merely as statistical errors, but as objects of qualitative investigation. This protocol evaluates the boundaries of artificial semantic reasoning through a newly proposed taxonomy of five agreement and divergence categories, structured to classify the dialectic intersections between human coders and the LLM. This workflow is visually mapped in Figure 1, detailing the pipeline from corpus preparation to interpretive curation.

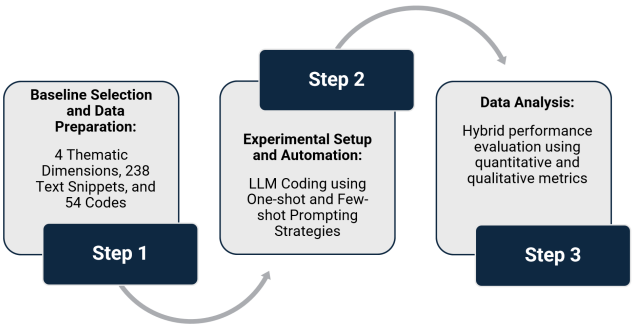

Figure 1: Methodological Workflow of the Study. Source: Created by the author.

To guide the execution of this study and evaluate the phenomenon of Human-AI collaborative coding, we formulated three core Research Questions (RQs). These map directly to Wohlin et al. [41]'s *Scoping* and *Planning* phases, each backed by a specific methodological rationale:

- **RQ1 (Scoping):** *What is the feasibility and accuracy of an LLM in the deductive coding of qualitative data in SE when compared against a human baseline?*
  Rationale: Establishes the baseline performance of the model in reproducing specialized human judgment, evaluating whether automated classification meets empirical SE standards.
- **RQ2 (Planning):** *How do different prompting strategies (One-shot vs. Few-shot) influence the reliability and consistency of the generated results?*
  Rationale: Investigates In-Context Learning (ICL) dynamics to identify whether additional context stabilizes the model's probabilistic nature or introduces disruptive noise.
- **RQ3 (Reporting):** *What application scenarios can guide the productive use of LLMs as support tools for qualitative analysis in Software Engineering?*
  Rationale: Defines the practical boundaries of the Human-in-the-Loop (HITL) paradigm to safely integrate LLMs into qualitative workflows.

3.1 Scoping and Data Preparation

To ground the evaluation, the research by Cerqueira et al. [10] on Empathy in SE was selected as the comparison Baseline. This choice is justified by the methodological robustness of the original work, whose codes and artifacts were peer-validated. Accordingly, the results consolidated by the researchers serve as the ground truth to measure the model's performance and interpretive sensitivity. We explicitly acknowledge an epistemological shift in this design. While the original baseline codes emerged through an inductive thematic analysis, our protocol purposely repurposes this peer-validated codebook as a static, closed-ended framework for subsequent deductive classification. Rather than attempting to replicate the human process of inductive conceptual discovery, this setup tests the LLM's capacity to execute a strict, rules-based mapping of specialized qualitative data under identical constraints. In doing so, it transforms a traditional qualitative artifact into a stable local benchmark for consistency auditing.

The corpus consists of 22 articles collected from the DEV online publishing platform, written by software professionals, totaling 24,565 words. For the experiment, 238 previously segmented and validated textual excerpts were used, organized into four thematic dimensions: **Meaning** (53 excerpts) focusing on conceptual definitions; **Barriers** (32 excerpts) mapping factors that hinder practice; **Practice** (76 excerpts) capturing concrete actions; and **Effects** (77 excerpts) evaluating observed outcomes.

The original codebook, consisting of 54 distinct codes and their respective definitions, was preserved in its entirety to ensure the fidelity of the deductive replication. The distribution of codes across the thematic dimensions is as follows: 5 codes for Meaning, 6 for Barriers, 15 for Practice, and 28 for Effects. In the Meaning dimension, codes include *Compassion*, *Emotional Sharing*, and *Understanding*; the latter defined as *"The capacity to understand another person's thoughts, needs, feelings, problems, and experience"*. In the Barriers dimension, which investigates obstacles to empathy in SE, codes such as *Individualistic Perspective*—defined as *"The attitude of some*

practitioners who prioritize their own methods and preferences over collaboration and support, lacking effort to get the perspective of others"—and *Work Constraints* are featured.

In the Practice dimension, codes include *Accepting and Giving Feedback* and *Adopting Good Programming Practices*, the latter defined as *"Thinking and caring about future code maintainers by following best programming practices"*. Finally, the Effects dimension addresses the impact of empathy on the well-being of SE professionals, with codes such as *Burnout*, *Productivity*, and *Human Connections*, defined as *"Stronger human connections and better interpersonal relationships"*. These examples illustrate how empathy is addressed within the context of SE in the work of Cerqueira et al. [10]. The complete codebook, containing definitions for all codes, can be consulted in the artifacts of this study (as indicated in the Artifact Availability section) or in the original study's repository [9].

3.2 Experimental Setup and Planning

In alignment with Wohlin et al. [41]'s planning phase, Gemini was adopted as the representative model for this experiment, specifically version Gemini 2.5 Pro (identified as the highest-reasoning version available during the experimental execution period). This choice is justified by its combination of high analytical performance with the availability of an Application Programming Interface (API) offering a free usage quota, ensuring operational feasibility and agility in the validation cycle.

The decision to focus the experiment on the Gemini model, rather than conducting a comparative analysis between multiple tools, is grounded in the volatility of the state-of-the-art in Generative AI. Given the frequency of updates, studies based purely on static benchmarking risk rapid obsolescence [11, 45]. Therefore, the objective of this work is not merely to measure the performance of a specific model, but to propose and validate a methodological framework that remains applicable regardless of the technological evolution of LLMs.

The interaction was automated using Python scripts and the Google Generative AI library. To ensure a realistic and accessible usage scenario, hyperparameters (*Temperature, Top-K, Top-P*) were kept at their default settings, thus evaluating the model's native capability to process the thematic domain.

3.3 Prompt Engineering

The development of the instructions followed an iterative refinement cycle. The final prompt architecture comprised six essential components: (1) Persona, (2) Research Question Contextualization, (3) Conceptual Codebook Definitions, (4) Scope Instructions, (5) Structured Output Format with mandatory justification, and (6) Examples. The full prompts can be consulted in the study's replication package, as detailed in the Artifact Availability Section.

Two strategies were compared: One-shot, which provides a single benchmark for format and analytical depth, and Few-shot, which supplies multiple real examples from the baseline (excluded from the final analysis) to enable In-Context Learning by comparison. Few-shot proved unfeasible for the Practice and Effects dimensions because they contain many codes (15 and 28, respectively) and the original excerpts often received multiple simultaneous codes. That made it impossible to extract isolated, unambiguous examples for each code—essential for constructing a clean instructional prompt—so Few-shot was applied only to the Meaning and Barriers dimensions.

3.4 Data Analysis

The evaluation of the model's capabilities followed a hybrid protocol. Proceeding from the premise that purely quantitative approaches fall short by treating interpretive divergences merely as statistical errors, the numerical assessment was complemented by a qualitative taxonomical analysis. To avoid ambiguity regarding standard computational resource metrics (such as inference execution time or token consumption costs), this study formally operationalizes the core concepts evaluated in our Research Questions as follows:

Classification Performance (Accuracy and Metric Adherence): Investigated primarily in RQ1 and RQ2, this dimension represents the model's strict adherence to the human baseline. It is quantitatively measured by the ratio of identical code assignments and targeted penalties for code omissions or background noise, as formulated through the Weighting System (Equation 1).

Methodological Feasibility: Investigated in RQ1 and RQ3, feasibility is operationalized as the LLM's capacity to process high-dimensional sociotechnical codebooks without incurring critical semantic collapse (hallucinations). This is evaluated qualitatively through the deterministic convergence of its structured justifications and its strict adherence to the abstract construct definitions provided.

In the quantitative phase, the operationalization was based on a weighting system that establishes the human Baseline as the reference for full accuracy (100%), aiming to measure the LLM's degree of adherence to the classifications validated by specialists, as described below:

(1) Human Reference (Baseline — B): For statistical calculation purposes, each code assigned by the human specialist was considered the correct target, receiving a weight of 1. Let N_B be the total number of codes present in the baseline for a given excerpt.

(2) Agreement Calculation (LLM Adherence — A): The model's score was calculated as the ratio between the number of matching codes (N_A) and the total number of codes in the baseline. This index reflects how closely the LLM aligned with the human reference, with penalties applied for omissions or for the generation of codes not present in the baseline (noise).

At the end of the process, the overall accuracy percentage of the LLM relative to the human classification was obtained. This index is given by the sum of weighted matching codes ($\sum N_A$) divided by the total sum of codes from the human baseline ($\sum N_B$), as expressed in Equation 1.

$$Accuracy\ Index(\%) = \frac{\sum N_A}{\sum N_B} \times 100 \tag{1}$$

Although the weighting system provided an initial statistical basis, purely quantitative metrics proved insufficient for handling multiclass classification and the interpretive nuances of the data. During the analysis, disagreement patterns were identified that revealed the probabilistic nature of the model: while frequently

identifying codes identical to the baseline, the LLM tended to generate supplementary suggestions that did not always constitute binary errors, but rather valid contextual perspectives.

To qualify these outputs before synthesizing the results, the divergences were subjected to triangulation between two researchers (the first author and another researcher experienced in qualitative analysis), who classified the responses into five Categories of Agreement and Divergence. This process was supported by the mandatory justifications generated by the model, ensuring the traceability of the AI's inferential reasoning. The resulting taxonomy, detailed in Table 1, allowed for a systematic distinction between accuracy failures and legitimate interpretive insights.

Table 1: Categories of Agreement and Divergence

Category	Criterion
1. Full Agreement	Human and LLM assign exactly the same code(s) to the text. Both codings are identical and correct.
2. Equivalent Coding with Expansion	Human and LLM agree on the core theme, enriched through complementary details that do not contradict the original baseline.
3. Valid Interpretive Divergence	Different codings, both correct and defensible. Human and LLM capture equally relevant facets without a clear winner.
4. Superior Human Coding	Human classification is more precise, complete, or relevant compared to the automated tool.
5. Superior LLM Coding	Automated classification is more precise or complete, identifying valid nuances that went unnoticed by the human.

It is important to note that the divergence categories in Table 1, especially Category 2 (Equivalent Coding with Expansion), do not reflect methodological errors, fatigue, or omissions in the human baseline. Rather, they arise from the epistemological difference between human interpretive parsimony and the LLM's literal exhaustiveness: human coders synthesized the core concept inductively, while the LLM performed strict deductive mapping, exhaustively matching any plausible code. Thus, these categories indicate a structural shift in coding behavior, not simple classification mistakes.

The consolidation of the data generated the definitive dataset for the statistical analysis and the subsequent qualitative interpretation, the results of which are detailed in the following section.

4 Results

The data analysis integrates performance metrics and interpretive patterns to answer the formulated RQs. This systematization of quantitative and qualitative findings underpins the discussion on the feasibility and limits of using LLMs in qualitative coding.

4.1 Accuracy Evaluation: Quantitative and Qualitative Perspectives

The analysis of the results began with the application of the Weighting System to measure the LLM's statistical performance relative to the human baseline. This quantitative approach allowed for the identification of accuracy levels based on Equation 1 across all thematic dimensions, as summarized in Table 2.

Table 2: LLM Classification Accuracy (Accuracy Index %) per Dimension and Strategy

Dimension	One-shot	Few-shot
Meaning	77.23%	76.33%
Barriers	68.91%	70.08%
Practice	45.18%	-
Effects	45.29%	-

The overall accuracy indices, calculated via Equation 1, showed similar performance for the two prompting techniques in dimensions where both could be compared. In the Meaning dimension, one-shot achieved 77.23% versus 76.33% for few-shot; in Barriers, the indices were 68.91% and 70.08%, respectively. Thus, providing multiple examples did not yield a meaningful improvement in model precision. For Practice and Effects—dimensions where only the one-shot strategy was technically feasible—accuracy was 45.18% and 45.29%, respectively. Although the weighting system offered an initial quantitative basis, it was insufficient to explain the divergences, particularly the proliferation of codes in dimensions with higher taxonomic complexity. This finding motivates the following interpretive analysis, which employs the taxonomy of divergences to analyze the model's behavior across each thematic dimension.

4.2 Thematic Dimension Analysis

The following analysis presents the performance of the one-shot and few-shot strategies in applicable dimensions, together with the one-shot results for the remaining dimensions. Each dimension is evaluated against the proposed taxonomy to characterize interpretive nuances and compare the model's rigor with the baseline.

4.2.1 Meaning Dimension.

The Meaning dimension comprised 53 excerpts focused on conceptual definitions of empathy. The LLM's performance revealed high alignment with the baseline: in the One-shot strategy, 84.91% of cases fell into agreement categories, consisting of 49.06% Full Agreement and 35.85% Equivalent Coding with Expansion (32.08% by the LLM and 3.77% by the human). In the Few-shot strategy, aggregate agreement fell to 75.00%, consisting of 45.83% Full Agreement and 29.17% Equivalent Coding with Expansion (27.08% by the LLM and 2.09% by the human).

Table 3: Frequency per Category in the Meaning Dimension

Category	One-Shot (N=53)	Few-Shot (N=48)
Total Agreement	26 (49.06%)	22 (45.83%)
Equiv. Coding (LLM Exp.)	17 (32.08%)	13 (27.08%)
Equiv. Coding (Human Exp.)	2 (3.77%)	1 (2.09%)
Superior Human Coding	8 (15.09%)	12 (25.00%)

The detailed analysis in Table 3 indicates that, although the Full Agreement rates are high in both strategies (49.06% for One-shot and 45.83% for Few-shot), the model's behavior varies significantly according to the support provided in the prompt. The error patterns and the expansion tendency observed in this dimension, as well as the implications of the Few-shot strategy, will be further explored in Section 5.

4.2.2 Barriers Dimension.

The Barriers dimension comprised 32 excerpts focused on the obstacles to empathetic practice in SE. The LLM's performance demonstrated consistent alignment with the baseline: in the One-shot approach, 75.00% of the cases fell into agreement categories, consisting of 40.63% Full Agreement and 34.37% Equivalent Coding with Expansion (28.12% by the LLM and 6.25% by the human). In the Few-shot strategy, the aggregate agreement was 69.23%, with 42.31% Full Agreement and 26.92% Equivalent Coding with Expansion (23.07% by the LLM and 3.85% by the human).

Table 4: Frequency per Category in the Barriers Dimension

Category	One-Shot (N=32)	Few-Shot (N=26)
Full Agreement	13 (40.63%)	11 (42.31%)
Equiv. Coding (LLM Exp.)	9 (28.12%)	6 (23.07%)
Equiv. Coding (Human Exp.)	2 (6.25%)	1 (3.85%)
Valid Interpretive Divergence	1 (3.12%)	1 (3.85%)
Superior Human Coding	7 (21.88%)	7 (26.92%)

The data presented in Table 4 reveal stability in the accuracy rates between the two approaches, suggesting that the addition of examples did not mitigate the model's limitations regarding interpretive nuances. The persistence of errors in passages using metaphors or depending on specific cultural contexts—elements that challenge the AI's abstraction capabilities—will be discussed in Section 5, aiming to define the scope of the model's autonomy in interpreting subjective phenomena.

4.2.3 Practice Dimension.

The Practice dimension comprised 76 excerpts and exhibited a high taxonomic density (15 codes). Due to this complexity and the predominance of multi-label classifications, the Few-shot strategy proved technically unfeasible, and the analysis focused on the One-shot strategy. In this dimension, the model achieved an aggregate agreement of 69.74%, comprising 11.84% Full Agreement and 57.90% Equivalent Coding with Expansion (56.58% by the LLM and 1.32% by the human).

Table 5: Frequency per Category in the Practice Dimension

Category	One-Shot (N=76)
Full Agreement	9 (11.84%)
Equiv. Coding (LLM Exp.)	43 (56.58%)
Equiv. Coding (Human Exp.)	1 (1.32%)
Valid Interpretive Divergence	8 (10.53%)
Superior Human Coding	12 (15.79%)
Superior LLM Coding	3 (3.95%)

Results for the Practice dimension indicate a shift in the model's behavior in high taxonomic density scenarios, characterized by a decrease in Full Agreement in favor of more fragmented coding (Equivalent Coding with Expansion). The predominance of expansions by the LLM (Equiv. Coding — LLM Exp.) points to the tool acting as an agent of semantic exhaustivity. This reading is reinforced by the emergence of cases where the model surpassed the level of detail provided by humans (Superior LLM Coding). These behavioral patterns, which redefine the interaction between the researcher and the AI, will be explored in Section 5.

4.2.4 Effects Dimension.

The Effects dimension comprised 77 excerpts and the densest taxonomy in the study, with 28 distinct codes. As in the Practice dimension, the analysis is based solely on the One-shot strategy. For this dimension, aggregate agreement reached 66.23%, reflecting the study's lowest Full Agreement rate (10.39%) and the highest incidence of Equivalent Coding with Expansion (55.84%; with 54.54% attributed to the LLM and 1.30% from the human).

Table 6: Frequency per Category in the Effects Dimension

Category	One-Shot (N=77)
Full Agreement	8 (10.39%)
Equiv. Coding (LLM Exp.)	42 (54.54%)
Equiv. Coding (Human Exp.)	1 (1.30%)
Valid Interpretive Divergence	21 (27.27%)
Superior Human Coding	4 (5.20%)
Superior LLM Coding	1 (1.30%)

The results of the Effects dimension confirm the model's profile in scenarios of broad taxonomic complexity, showing the highest incidence of Valid Interpretive Divergences (27.27%) and semantic expansions (Equivalent Coding with Expansion, both LLM and Human) in the study. The reduction in critical failures (Superior Human Coding), combined with the model's tendency to act as an agent of exhaustivity (Expansion), suggests that the LLM's utility lies more in its scanning and verification capability than in the exact replication of human judgment.

5 Discussion

The joint analysis of the four dimensions reveals that the LLM's behavior is not erratic but governed by systematic patterns of interaction with the codebook and the text.

5.1 Consistency and Traceability

Supporting prior work on generative AI's stabilization capacity [4, 7, 38], the experiments showed strong deterministic convergence. Even across prompting variations (one-shot vs. few-shot) in independent processing instances, the model consistently used the same logical bases and semantically equivalent justifications. For example, in the Meaning dimension (excerpt P31, full agreement), the model grounded the categories Understanding and Perspective Taking with nearly identical logical structures:

> **Prompt One-Shot:** *"The snippet highlights the need to "understand the needs of the users," which directly maps to the 'Understanding' code. It also explicitly describes the action of "see[ing] the app from our users perspective," which is a clear instance of 'Perspective Taking'".*
>
> **Prompt Few-Shot:** *"The snippet explicitly mentions the need to "understand the needs of the users," which aligns with the 'Understanding' code. It also directly describes the action to "see the app from our users perspective," which is a textbook example of 'Perspective Taking'".*

However, this logical stability should not be confused with infallibility. The cross-sectional analysis reveals that, while consistent, the LLM manifests systematic interpretive tendencies that distinguish it from human judgment. The following section explores these patterns, highlighting how the interaction between algorithmic logic and qualitative subjectivity shapes the coding process.

5.2 LLM Behavioral Patterns

Three core dynamics were identified that define the collaboration between the researcher and the model:

5.2.1 Excessive Granularity Bias (The Exhaustive Agent).

A consistent pattern emerged: the model prioritizes exhaustivity over interpretive parsimony. While the human coder tends to select the core code, the LLM maps every detectable semantic nuance. In dimensions with high taxonomic density, such as Practice (15 codes) and Effects (28 codes), the Equivalent Coding with Expansion rate by the LLM reached 56.58% and 54.5%, respectively. This phenomenon highlights a high scanning capacity. But it also reconfigures the researcher's role to that of a curator, responsible for validating whether such expansions represent analytical enrichment or noise. Such behavior corroborates the propensity of LLMs toward "over-segmentation" and the generation of more granular categories than those produced by humans [4, 8]. However, algorithmic exhaustivity requires caution; as Kapania et al. [24] warn, it can generate "infinite details without depth" or, as pointed out by Lee et al. [26], "repetitions that inadequately capture the content". In this context, human curation becomes indispensable for filtering the actual theoretical contribution [1, 35, 43].

Key Finding 1 (Excessive Granularity): *In high-dimensional taxonomies, the LLM operates as an exhaustive scanning agent. Rather than a performance error, this over-segmentation shifts the researcher's role from a manual coder to an inference curator.*

This pattern is unlikely to be a statistical coincidence. This scanning capacity highlights the LLM's efficiency in handling the operational effort of exhaustive mapping, reinforcing its strategic utility in three methodological scenarios supported by contemporary literature:

Consolidation Support (Inductive Methods): The model acts as an additional coder for triangulation purposes. This scenario confirms the propositions of Bennis and Mouwafaq [6], Lee et al. [26], who recommend treating the LLM as a complementary member of the research team capable of providing "automated triangulation". Similarly, Bijker et al. [8], Tai et al. [38] corroborate the idea of using LLMs as a "second coder". It should be noted, however, that this practice is not without criticism. Authors such as Friedman et al. [17] refute the validity of human-AI triangulation, arguing that true investigative triangulation requires an epistemological awareness that algorithmic models lack.

Scalability (Deductive Methods): The LLM enables the analysis of large data volumes by ensuring that multiple facets of the text are captured systematically. This finding corroborates the points made by Barros et al. [4] regarding the superiority of AI in scaling the analysis of unstructured data. Specifically in deductive approaches, the model's performance confirms the success reported by Xiao et al. [42] in using predefined codebooks to categorize vast datasets in a systematic and agile manner.

Consistency Auditing: The use of the model mitigates the human bias of omission, identifying elements that might be neglected due to fatigue or selective attention. This utility directly confirms the observations of Bijker et al. [8], who highlight the advantage of AI in not suffering from "cognitive fatigue". This lets it identify subtle connections that human inattention might ignore. Furthermore, as observed by Tai et al. [38], this algorithmic constancy acts as a valuable analytical validation tool against potential researcher biases.

5.2.2 The Limit of Subjectivity (Inexplicable Deduction).

The model's failures—categorized as Superior Human Coding—ranged from 5.20% in Effects to 26.92% in Barriers, and occurred mainly in interpretations requiring reflexive analogies or tacit knowledge. A tendency toward literalism was especially evident in Barriers, where metaphors and culturally dependent expressions exceeded the model's abstraction capabilities. Unable to link broad social issues to concrete software-industry obstacles, the model assigned no code ("None") to excerpt P27. This behavior shows that contextual sensitivity for interpreting implicit meaning remains a human competence. Below is the original excerpt and the model's justification for the omission:

> **Excerpt (P27) - Barriers Dimension:** *"My personal conclusion is that the most common underlying problem for many of the issues we try to solve in our industry are related to the lack of empathy. It's not hard to notice how that works every day, from the politician who asks for money to build a wall to stop immigration, to the lady who yells at a waitress without any effort to feel a little bit of empathy for someone who might be having a bad day. An entire society not genuinely understanding what other person feels."*
>
> **Model Justification:** *"This text provides a general commentary on the societal lack of empathy, using examples outside the software development context. It does not describe a specific barrier within a software engineering team or organization that maps to the provided codes."*

The analysis of this case highlights a critical limitation: the model presents a logically coherent and, at first glance, "convincing" justification by classifying the examples as decontextualized; however, the analysis proves to be superficial. The LLM fails by neglecting the reflexive analogy that associates the "underlying problem of the industry" with the social dynamics described in the text. This behavior demonstrates that the model is capable of generating articulate explanations. Yet the sensitivity to interpret implicit contexts and mobilize tacit knowledge remains, invariably, a competence of the human researcher.

This empirical limitation finds support in the literature. De Paoli [14] argues that LLMs operate fundamentally from a structural and probabilistic perspective, focusing on the form of language to the detriment of its meaning. Since they lack social experience or a conception of the world, their analysis becomes a statistical correspondence that lacks human semantic depth.

Building on this point, Tai et al. [38] warn that the model's understanding is restricted to the patterns present in its training data. This disconnection is detailed by Kapania et al. [24], who observe a lack of "palpability" in the models' responses. Although they generate coherent text, LLMs do not possess the embodied understanding necessary to produce knowledge grounded in emotions and lived experience.

Such a gap is confirmed by the systematic mapping of Barros et al. [4], who identified significant restrictions of LLMs in interpreting complex meanings and a lack of sensitivity to emotional context. Consequently, there is a tangible risk of losing nuances or

potentially interesting information that would be readily noticed by human analysts due to their contextual sensitivity [35].

Key Finding 2 (Subjectivity Boundaries): *Textual contexts that rely on reflexive analogies, metaphors, or tacit knowledge tend to trigger classification omissions by the LLM. The model's semantic boundaries depend heavily on literal text structures, making human interpretive authority indispensable.*

5.2.3 Potential for Discovery and Triangulation.

In dimensions marked by a high interpretive load, the occurrence of Valid Interpretive Divergences and cases of Superior LLM Coding redefine the tool's role. The model transcends automation to establish itself as a triangulation agent, acting on two fronts: the proposition of complementary perspectives and the surpassing of human precision.

Case A: Interpretive Richness (Valid Divergence)

When a text allows multiple defensible readings, a divergence can be valid rather than erroneous. In excerpt P13 (Practice), the human coder chose Understanding the Stakeholders (subjective understanding), while the LLM selected Considering Different Perspectives (acceptance of distinct mental models); both are grounded in the account of varied technical backgrounds. This supports using the model as a "second opinion" to challenge confirmation bias and broaden theoretical saturation.

> **Excerpt (P13) - Practice Dimension:** *"I have worked as a software developer for a few years now, and the most common thing that I had to deal with, was figure how works some code that another dev wrote. Each person has your own backgrounds, convictions, policies and mental model, as long that their code follows the team/company conventions and not present some real issue there is nothing to worry about."*
>
> **Model Justification:** *"The text acknowledges that every developer has their own "backgrounds, convictions, policies and mental model," showing an explicit understanding and acceptance of different viewpoints. This directly reflects the practice of 'Considering Different Perspectives.'"*

This empirical finding aligns directly with the premise of Lee et al. [26] that there is no "absolute truth" in qualitative analysis. From this perspective, interpretive divergences between human and machine are not necessarily failures, but rather indicators that the data is being examined from complementary angles. Recent literature corroborates that treating the LLM as a "second coder" fosters the reflexivity of the human analyst [8, 38]. By proposing alternative interpretations, the model acts as a mechanism to mitigate bias and challenge the researcher's preconceived interpretive frameworks [38].

Case B: Exceeding Human Precision (Superior LLM Coding)

In this case, the model identified explicit elements overlooked by the human coder, mitigating omission bias. In excerpt P16 (Practice), the human coding was more general, while the LLM applied the codebook with greater rigor. The model identified the phrase "put yourselves in the shoes" as Embodying the Stakeholders, a technically more specific choice than the human's Understanding the Stakeholders.

> **Excerpt (P16) - Practice Dimension:** *"[...] It's safe to consider it reality that other engineers in your organization are going to disagree with you. [...] Make every effort to understand your disagreeing coworker's point of view. [...] When someone makes a request or asks you for something, put yourselves in the shoes of someone who doesn't know all the terminology [...]. Above all, remember that you're all in pursuit of the same goals."*
>
> **Model Justification:** *"The text advocates for "Considering Different Perspectives" by urging developers to understand a disagreeing coworker's point of view. It describes "Accepting and Giving Feedback" through the lens of offering praise to make constructive feedback more effective. It explicitly promotes "Embodying the Stakeholders" by telling developers to put themselves in the shoes of a non-technical person. Finally, it encourages "Being Mindful" by reminding developers of their shared goals during interactions."*

The analysis reveals robust logical connections that were overlooked in the manual coding. By identifying valid divergences and correcting omissions, the model assumes the role of a second coder [8, 38]. This capability ensures that relevant nuances of the corpus are not overlooked due to fatigue or selective attention [8]. This consolidates the LLM as a supporting tool for consistency auditing in qualitative research.

Key Finding 3 (Discovery & Triangulation): *Interpretive divergences between human and algorithmic agents frequently represent valid alternative perspectives rather than statistical errors. The LLM acts as a robust dialogic agent for consistency auditing, successfully mitigating human omission bias.*

5.3 Answers to Research Questions

This section synthesizes the study's main findings in light of the proposed Research Questions (RQ). The discussion articulates quantitative performance results with the qualitative analysis of the model's behavioral patterns.

5.3.1 Feasibility and Classification Accuracy (RQ1).

These findings confirm technical feasibility. It demonstrated a high scanning and exhaustivity capacity, but lower performance in capturing subjective nuances and implicit contexts. Feasibility rests not on replacing the researcher, but on the LLM's role as a co-coder under supervision. Table 7 synthesizes the fundamental distinctions observed.

Table 7: Functional Contrast: LLM vs. Human Researcher

Aspect	LLM (Algorithmic)	Human (Semantic)
Scanning	Exhaustive: maps all probable associations.	Selective: focuses on core relevance and intentionality.
Nuances	Limited: dependent on the literalness of the text.	Deep: based on context and tacit knowledge.
Role	Proposer of granularity and omission auditor.	Curator, final judge, and interpreter of subjectivities.

5.3.2 Prompting Strategy and Reliability (RQ2).

The One-shot strategy (focused on conceptual definition) proved to be superior and more robust than the Few-shot approach. It was

observed that providing examples can induce overfitting, where the model replicates superficial patterns from the examples instead of applying the abstract rule of the code. This phenomenon was clearly observed in the Meaning dimension, where the use of examples (Few-shot) increased the failure rate from 15.09% to 25.00%, suggesting that superficial similarity with the examples overrode the conceptual definition of the codebook. In addition, the mandatory requirement for structured justifications proved essential for traceability, allowing for the auditing of the algorithmic logic.

5.3.3 Guidelines and Application Scenarios (RQ3).

Based on the qualitative analysis of the interpretive intersections between the model and the baseline, three potential usage scenarios for integrating LLMs into qualitative workflows were identified: (i) Triangulation in inductive methods, acting as an additional coder; (ii) Scalability in deductive methods, enabling the systematic processing of large data volumes; and (iii) Consistency auditing, helping to reduce omissions bias. The core insight derived from these scenarios is the researcher's transition from the role of coder to that of Curator, responsible for validating the model's excessive granularity and converting noise into insight.

6 Threats to Validity

Despite providing evidence regarding the feasibility and interaction patterns of LLMs in qualitative data coding, the validity of this study is subject to limitations. These threats, as well as the strategies adopted to mitigate them, are detailed below.

Single-Model Scope (Internal Validity): This work examined a single LLM (Gemini 2.5 Pro). The behavioral patterns identified may not generalize to other models with different architectures, training corpora, or inference strategies. Future work should replicate the protocol with alternative models to assess cross-model consistency.

Model Volatility (External Validity): Continuous model updates by vendors may impact long-term reproducibility. We address this volatility by framing our contributions around stable qualitative behavioral patterns and a transferable methodological protocol rather than transient performance benchmarks.

Domain Generalization (External Validity): Conducting this evaluation within a single sociotechnical domain (Empathy in SE) limits the generalizability of the findings to other qualitative contexts. However, we present the study as a transferable empirical benchmark, contributing reusable protocols to the qualitative research community in software engineering.

Prompt Bias (Conclusion Validity): The sensitivity of LLMs to the lexicon of instructions can implicitly bias responses. To ensure that variations in results stemmed from the content of the excerpts rather than oscillations in the instructions, rigid and standardized templates were adopted for all interactions, which are available in this study's replication package.

Prompting Infeasibility (Internal/Conclusion Validity): High-dimensional taxonomies (15 and 28 codes) with multi-label characteristics made extracting isolated, representative few-shot examples for the Practice and Effects dimensions methodologically unviable. We mitigated this by enforcing a one-shot approach as a universal baseline across all dimensions, preserving formatting stability and isolating the model's abstract reasoning under identical protocol constraints.

Codebook Inheritance and Construct Validity: A construct threat arises from using Cerqueira et al.'s inductively derived codebook as a fixed benchmark for automated deductive coding. This setup creates tension between human inductive code generation and LLM deductive assignment, which can explain observed divergences in granularity since the model is tested against a framework produced from the same transcripts. We mitigate this by framing the LLM as an automated consistency auditor under strict, rules-based constraints, not as a substitute for human inductive discovery.

Human Evaluation Subjectivity: Mitigating researcher subjectivity in taxonomy assignment was achieved through independent classification by two investigators. A subsequent consensus cycle resolved all discrepancies, ensuring a consolidated interpretation.

Technical Scope: We deliberately avoided fine-tuning techniques to ensure out-of-the-box reproducibility. Relying strictly on public APIs and prompt engineering guarantees accessibility for the broader empirical research community.

7 Conclusion

The present work evaluated the technical and methodological feasibility of using LLMs for the deductive coding of qualitative data in SE. Through an experiment focused on the theme of empathy in SE, it was observed that the introduction of language models does not represent a mere automation of mechanical tasks, but rather a reconfiguration of the traditional analytical process.

Our core findings demonstrate that LLM feasibility for deductive qualitative coding remains strictly conditioned on human curation. It excels as an exhaustive scanning agent, effectively mitigating human omission bias due to fatigue. However, human interpretive authority remains irreplaceable for capturing tacit contexts and reflexive analogies. Methodologically, the clear conceptual framing of the One-shot strategy outperformed the Few-shot approach by preventing context-level overfitting and ensuring higher analytical stability.

As future work, we intend to replicate this protocol in a second qualitative study in SE to evaluate its transferability. The findings from both studies, including the systematization of the practical scenarios identified in this phase, will be consolidated into a definitive version of guidelines for deductive coding with LLMs, delivering to the community a mature artifact validated in multiple contexts.

Artifact Availability

The experimental materials produced in this study (automation scripts, prompt templates, coded datasets, and detailed analysis records) are available for auditing and replication at [30].

Acknowledgements

This work was partially supported by UEFS-AUXPPG 2023/2024/2025 and CAPES-PROAP 2023/2024/2025 grants.

References

[1] Anas Al-Fattal and Jasvir Singh. 2025. Comparative Reflections on Human-Driven and Generative Artificial Intelligence-Assisted Thematic Analysis: A Collaborative Autoethnography. International Journal of Qualitative Methods 24 (2025), 16094069251337870. doi:10.1177/16094069251337870

[2] Jothan Almeida. 2025. Prompt Engineering: A Comparative Study of Prompting Techniques in AI Language Models. In 2025 IEEE Integrated STEM Education Conference (ISEC). IEEE, 1–4. doi:10.1109/ISEC64801.2025.11147384

[3] Muneera Bano, Rashina Hoda, Didar Zowghi, and Christoph Treude. 2023. Large language models for qualitative research in software engineering: exploring opportunities and challenges. Automated Software Engineering 31, 1 (2023), 8. doi:10.1007/s10515-023-00407-8

[4] Cauã Ferreira Barros, Bruna Borges Azevedo, Valdemar Vicente Graciano Neto, Mohamad Kassab, Marcos Kalinowski, Hugo Alexandre D. Do Nascimento, and Michelle C.G.S.P. 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). 48–55. doi:10.1109/WSESE66602.2025.00015

[5] Martin W Bauer and George Gaskell. 2017. Pesquisa qualitativa com texto, imagem e som: um manual prático. Editora Vozes Limitada.

[6] Issam Bennis and Safwane Mouwafaq. 2025. Advancing AI-driven thematic analysis in qualitative research: a comparative study of nine generative models on Cutaneous Leishmaniasis data. BMC Medical Informatics and Decision Making 25, 1 (2025), 1–14. doi:10.1186/s12911-025-02961-5

[7] Gaurav Beri and Vaishnavi Srivastava. 2024. Advanced Techniques in Prompt Engineering for Large Language Models: A Comprehensive Study. In 2024 IEEE 4th International Conference on ICT in Business Industry Government (ICTBIG). IEEE, 1–4. doi:10.1109/ICTBIG64922.2024.10911672

[8] Rimke Bijker, Nicki A Merkouris, Stephanie Sand Dowling, and Simone N Rodda. 2024. ChatGPT for Automated Qualitative Research: Content Analysis. J Med Internet Res 26 (25 Jul 2024), e59050. doi:10.2196/59050

[9] Lidiany Cerqueira. 2025. Experimental Package: Exploring Empathy in Software Engineering Articles. doi:10.5281/zenodo.15800354

[10] Lidiany Cerqueira, João Pedro Bastos, Danilo Neves, Glauco de Figueiredo Carneiro, Rodrigo, Sávio Freire, Jose Amancio Macedo Santos, and Manoel Mendonça. 2026. Exploring Empathy in Software Engineering: Insights from a Grey Literature Analysis of Practitioners' Perspectives. ACM Trans. Softw. Eng. Methodol. 35, 5, Article 136 (April 2026), 45 pages. doi:10.1145/3748721

[11] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2024. A Survey on Evaluation of Large Language Models. ACM Trans. Intell. Syst. Technol. 15, 3, Article 39 (March 2024), 45 pages. doi:10.1145/3641289

[12] Kathy Charmaz. 2009. A construção da teoria fundamentada: guia prático para análise qualitativa. Bookman Editora.

[13] John W Creswell and J David Creswell. 2021. Projeto de pesquisa: Métodos qualitativo, quantitativo e misto. Penso Editora.

[14] 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. doi:10.1177/08944393231220483

[15] Torgeir Dingsøyr, Phillip Schneider, Gunnar Rye Bergersen, and Yngve Lindsjørn. 2024. Challenges in Understanding the Relationship between Teamwork Quality and Project Success in Large-Scale Agile Projects. In Proceedings of the 2024 IEEE/ACM 17th International Conference on Cooperative and Human Aspects of Software Engineering (Lisbon, Portugal) (CHASE '24). Association for Computing Machinery, New York, NY, USA, 51–56. doi:10.1145/3641822.3641868

[16] Christian Djeffal. 2025. Reflexive Prompt Engineering: A Framework for Responsible Prompt Engineering and AI Interaction Design. In Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency (FAccT '25). Association for Computing Machinery, New York, NY, USA, 1757–1768. doi:10.1145/3715275.3732118

[17] Carli Friedman, Aleksa Owen, and Laura VanPuymbrouck. 2025. Should ChatGPT help with my research? A caution against artificial intelligence in qualitative analysis. Qualitative Research 25, 5 (2025), 1062–1088. doi:10.1177/14687941241297375

[18] Jie Gao, Kenny Tsu Wei Choo, Junming Cao, Roy Ka-Wei Lee, and Simon Perrault. 2023. CoAIcoder: Examining the Effectiveness of AI-assisted Human-to-Human Collaboration in Qualitative Analysis. ACM Trans. Comput.-Hum. Interact. 31, 1, Article 6 (Nov. 2023), 38 pages. doi:10.1145/3617362

[19] Hashini Gunatilake, John Grundy, Rashina Hoda, and Ingo Mueller. 2024. The impact of human aspects on the interactions between software developers and end-users in software engineering: A systematic literature review. Information and Software Technology 173 (2024), 107489. doi:10.1016/j.infsof.2024.107489

[20] Adam S Hayes. 2025. "Conversing" With Qualitative Data: Enhancing Qualitative Research Through Large Language Models (LLMs). International Journal of Qualitative Methods 24 (2025), 16094069251322346. doi:10.1177/16094069251322346

[21] Rashina Hoda. 2022. Socio-Technical Grounded Theory for Software Engineering. IEEE Transactions on Software Engineering 48, 10 (2022), 3808–3832. doi:10.1109/TSE.2021.3106280

[22] Xinyi Hou, Yanjie Zhao, Yue Liu, Zhou Yang, Kailong Wang, Li Li, Xiapu Luo, David Lo, John Grundy, and Haoyu Wang. 2024. Large Language Models for Software Engineering: A Systematic Literature Review. ACM Trans. Softw. Eng. Methodol. 33, 8, Article 220 (Dec. 2024), 79 pages. doi:10.1145/3695988

[23] Shivya Jyoti, Moulik Tejpal, and Jothi K R. 2025. Optimizing Generative AI Applications: A Comparative Study of Effective Prompting Techniques. In 2025 5th International Conference on Pervasive Computing and Social Networking (ICPCSN). 389–396. doi:10.1109/ICPCSN65854.2025.11035207

[24] Shivani Kapania, William Agnew, Motahhare Eslami, Hoda Heidari, and Sarah E Fox. 2025. Simulacrum of Stories: Examining Large Language Models as Qualitative Research Participants. In Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25). Association for Computing Machinery, New York, NY, USA, Article 489, 17 pages. doi:10.1145/3706598.3713220

[25] Mohammad Amin Kuhail, Jose Berengueres, Fatma Taher, Sana Zeb Khan, and Ansah Siddiqui. 2024. Designing a Haptic Boot for Space With Prompt Engineering: Process, Insights, and Implications. IEEE Access 12 (2024), 134235–134255. doi:10.1109/ACCESS.2024.3449396

[26] V Vien Lee, Stephanie CC van der Lubbe, Lay Hoon Goh, and Jose Maria Valderas. 2024. Harnessing ChatGPT for thematic analysis: Are we ready? Journal of Medical Internet Research 26 (31 May 2024), e54974. doi:10.2196/54974

[27] Per Lenberg, Robert Feldt, Lucas Gren, Lars Göran Wallgren Tengberg, Inga Tidefors, and Daniel Graziotin. 2024. Qualitative software engineering research: Reflections and guidelines. Journal of Software: Evolution and Process 36, 6 (2024), e2607. doi:10.1002/smr.2607

[28] Kashumi Madampe, Rashina Hoda, and John Grundy. 2024. Supporting Emotional Intelligence, Productivity and Team Goals while Handling Software Requirements Changes. ACM Trans. Softw. Eng. Methodol. 33, 6, Article 153 (June 2024), 38 pages. doi:10.1145/3664600

[29] Ggaliwango Marvin, Nakayiza Hellen, Daudi Jjingo, and Joyce Nakatumba-Nabende. 2024. Prompt engineering in large language models. In Data Intelligence and Cognitive Informatics. Springer Nature Singapore, Singapore, 387–402.

[30] Samuel Mercês. 2026. Experimental Package: Investigating the use an LLM in Deductive Coding within the Context of Software Engineering. doi:10.5281/zenodo.19651326

[31] Rudolf Messner, Samuel Smith, and Carol Richards. 2025. Artificial Intelligence and Qualitative Data Analysis: Epistemological Incongruences and the Future of the Human Experience. International Journal of Qualitative Methods 24 (2025), 16094069251371481. doi:10.1177/16094069251371481

[32] Mohaimenul Azam Khan Raiaan, Md. Saddam Hossain Mukta, Kaniz Fatema, Nur Mohammad Fahad, Sadman Sakib, Most Marufatul Jannat Mim, Jubaer Ahmad, Mohammed Eunus Ali, and Sami Azam. 2024. A Review on Large Language Models: Architectures, Applications, Taxonomies, Open Issues and Challenges. IEEE Access 12 (2024), 26839–26874. doi:10.1109/ACCESS.2024.3365742

[33] Luz Marcela Restrepo-Tamayo, Gloria Piedad Gasca-Hurtado, and Johnny Valencia-Calvo. 2024. Characterizing Social and Human Factors in Software Development Team Productivity: A System Dynamics Approach. IEEE Access 12 (2024), 59739–59755. doi:10.1109/ACCESS.2024.3388505

[34] Beatriz Santana, Lidivânio Monte, Bianca Santana de Araújo Silva, Glauco Carneiro, Sávio Freire, José Amancio Macedo Santos, and Manoel Mendonça. 2025. Psychological safety in software workplaces: A systematic literature review. Information and Software Technology 187 (2025), 107838. doi:10.1016/j.infsof.2025.107838

[35] Hope Schroeder, Marianne Aubin Le Quéré, Casey Randazzo, David Mimno, and Sarita Schoenebeck. 2025. Large Language Models in Qualitative Research: Uses, Tensions, and Intentions. In Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25). Association for Computing Machinery, New York, NY, USA, Article 481, 17 pages. doi:10.1145/3706598.3713120

[36] Carolyn B. Seaman, Rashina Hoda, and Robert Feldt. 2025. Qualitative Research Methods in Software Engineering: Past, Present, and Future. IEEE Transactions on Software Engineering 51, 3 (2025), 783–788. doi:10.1109/TSE.2025.3538751

[37] Robert E Stake. 2016. Pesquisa qualitativa: estudando como as coisas funcionam. Penso Editora.

[38] Robert H Tai, Lillian R Bentley, Xin Xia, Jason M Sitt, Sarah C Fankhauser, Ana M Chicas-Mosier, and Barnas G Monteith. 2024. An examination of the use of large language models to aid analysis of textual data. International

Journal of Qualitative Methods 23 (2024), 16094069241231168. doi:10.1177/16094069241231168

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. 30 (2017). https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[40] Zichong Wang, Zhibo Chu, Thang Viet Doan, Shiwen Ni, Min Yang, and Wenbin Zhang. 2024. History, development, and principles of large language models: an introductory survey. AI and Ethics (2024), 1–17. doi:10.1007/s43681-024-00583-7

[41] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, Anders Wesslén, et al. 2012. Experimentation in software engineering. Vol. 236. Springer.

[42] 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

[43] Lixiang Yan, Vanessa Echeverria, Gloria Milena Fernandez-Nieto, Yueqiao Jin, Zachari Swiecki, Linxuan Zhao, Dragan Gašević, and Roberto Martinez-Maldonado. 2024. Human-AI Collaboration in Thematic Analysis using ChatGPT: A User Study and Design Recommendations. In Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (Honolulu, HI, USA) (CHI EA '24). Association for Computing Machinery, New York, NY, USA, Article 191, 7 pages. doi:10.1145/3613905.3650732

[44] Robert K Yin. 2016. Pesquisa qualitativa do início ao fim. Penso Editora.

[45] He Zhang, Chuhao Wu, Jingyi Xie, Yao Lyu, Jie Cai, and John M. Carroll. 2025. Harnessing the power of AI in qualitative research: Exploring, using and redesigning ChatGPT. Computers in Human Behavior: Artificial Humans 4 (2025), 100144. doi:10.1016/j.chbah.2025.100144