Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 280 (2026) 552–559

Procedia Computer Science

www.elsevier.com/locate/procedia

The 17th International Conference on Ambient Systems, Networks and Technologies (ANT)
April 14-16, 2026, Istanbul, Türkiye

A Resource-Efficient Codebook-Driven Semantic Structuring Pipeline for Human-AI Dialogue in Ambient Intelligent Systems

Aisvarya Adeseye[a,*], Jouni Isoaho[a], Seppo Virtanen[a], Tahir Mohammad[a]

[a]*Department of Computing, University of Turku, Vesilinnantie 5, Turku 20014, Finland*

Abstract

Human–AI dialogue in ambient intelligent systems is increasingly relying on large language models (LLMs). When questions are generated dynamically to enable personalized and context-aware interactions, variations in phrasing and topical focus exist between conversations. Without structured organization, which is often extremely resource-intensive, conversational data remains fragmented and cannot be reliably used for systematic analysis or reporting. This study proposes a semantic structuring pipeline to map LLM-generated questions to shared codes, sub-themes, and themes using a predefined codebook. This multi-stage pipeline applies semantic screening, factor-based scoring, mathematical aggregation, and validation checks, supported by locally deployed LLMs and manual confirmation. The pipeline was evaluated on 6,030 question–response pairs collected from dynamic interviews across three research objectives. The framework achieved an overall mapping accuracy of 97% while reducing hallucinated semantic matches to 1.2% through layered validation. The results indicate that the framework effectively reduces hallucinated matches and improves mapping accuracy while remaining computationally efficient for private local deployment.

Keywords: Semantic Structuring Pipeline; Human–AI Dialogue; Large Language Models (LLMs); Codebook-Based Thematic Mapping; Resource-Efficient Local Deployment; Hallucination Mitigation; Ambient Intelligent Systems; Qualitative Data Analysis Automation

1. Introduction

Human–AI dialogue in ambient intelligent systems is increasingly depending on large language models (LLMs) [1]. When LLMs generate dynamic questions that support personalized and context-aware interaction that improve conversational flow, they create variations in wording and topic focus across conversations. Often, similar questions are phrased differently or address overlapping themes inconsistently [2]. As a result, responses become fragmented and are difficult to organize for systematic study. Without structured organization, large conversational datasets cannot be reliably analyzed or reported [3]. Manual coding is possible but becomes time-consuming and costly as the data volume increases [4]. Automated methods, such as keyword-based text classification, rule-driven coding tools, and

* Corresponding author
E-mail address: aisvarya.a.adeseye@utu.fi

direct LLM-assisted thematic analysis [5, 6], can significantly improve processing speed; however, without structured validation, they may introduce semantic misclassification, boundary confusion, or hallucinated topic alignments [7]. This creates the need for a resource-efficient [8] and accurate structuring pipeline that maintains analytical reliability while supporting privacy-preserving data processing.

Therefore, this study proposes a semantic structuring pipeline for systematically mapping LLM-generated questions using a four-level codebook hierarchy: Objective, Theme, Sub-theme, and Code. To achieve this, the study addresses two main objectives:

- **Objective 1:** To introduce a multi-stage semantic structuring pipeline that analyzes large volumes of dynamically generated conversational data.
- **Objective 2:** To evaluate the effectiveness and efficiency of the proposed pipeline by analyzing mapping accuracy, hallucination reduction, and computational complexity under local deployment conditions.

2. Related Work

Recent studies have increasingly explored the role of AI and LLMs in qualitative analysis and human–AI collaboration. However, focusing on assisting manual coding or interpretation rather than solving the problem of dynamically structuring conversational questions generated by LLMs. Yan et al. [9] examined how ChatGPT supports thematic analysis through human–AI collaboration. Their study showed that LLMs help researchers explore transcripts and accelerate coding tasks. The authors raised concerns about trust, reliability, and validation, while also echoing the need for transparent design and strong human oversight. Cheng et al. [10] conducted a systematic review and taxonomy of human–AI interaction workflows for text generation, including guiding, selecting, editing, and co-writing models, focusing on the creation of interactive designs. Bunt and Petukhova [11] focused on semantic and pragmatic precision in conversational AI systems and applied dialogue-act annotation standards, such as ISO 24617-2 to achieve structured interaction modeling. This methodology advances semantic precision but relies on formal dialogue tagging and annotated corpora. Moreover, Costa et al. [12] conceptualized AI as a co-researcher for qualitative workflows by introducing an AI model for collaborative thematic interpretation supported by structured prompting. Their work advances epistemological collaboration but remains focused on assisting with coding and theme development.

Nguyen and Welch [13] critically evaluated the application of Generative AI to qualitative data analysis. They warned of epistemic risks such as hallucinations, lack of transparency, and weak validation. Their findings emphasized the need for rigorous verification frameworks, which directly supported and motivated the validation-driven semantic structuring pipeline proposed in this study. Contrary to existing studies that primarily focus on manual thematic analysis or improving dialogue modelling, our study contributes a dedicated semantic structuring pipeline for dynamically mapping created LLM questions to predefined codes and themes at scale.

3. Methodology

This study proposes a semantic structuring pipeline shown in Fig. 1 to map conversation questions to predefined codes, sub-themes, and themes contained in a structured codebook. The data log questions were not manually mapped to the code by the researchers. Instead, each question was automatically evaluated against the codes in the codebook. Initially, a semantic screening gate checks whether a question has a clear relationship with the paired code. Only questions and code pairs that have full or partial relevance move to factor-based scoring. This mathematically aggregated the five key factors: concept clarity, explicitness, scope, emotional relevance, and boundary fit to produce an overall alignment score between each question and code. Based on this score, the questions were classified as having primary, secondary, or no thematic match. Finally, boundary, contrast, and negation checks were applied to confirm consistency and accuracy. Moreover, the validated code matches were grouped into sub-themes and themes to generate the final Question–Theme Map. Throughout the process (semantic screening gate, factor-based scoring, checks, and validation), manual validation checks were conducted to verify the automated results. Only manually confirmed outputs are moved to the next stage. More details about this process can be found in Subsections 3.1 and 3.2.

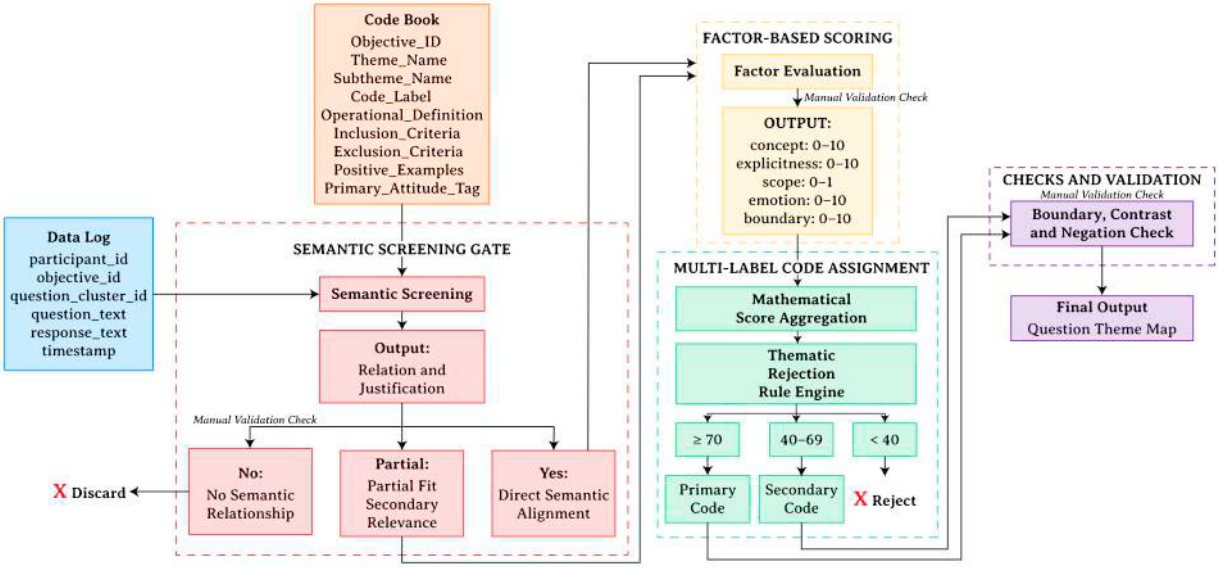

Fig. 1. Semantic Structuring pipeline for Mapping Questions to Themes through codes Using Factor-Based Scoring

3.1. Semantic Structuring Pipeline

The pipeline integrates five main components: **Inputs**, **Semantic Screening Gate**, **Factor-based Scoring**, **Multi-Level Code Assignment**, and **Checks and Validation**. Each stage contributes to building reliable mappings between the LLM-generated questions and the predefined themes. The following subsections describe these components in detail.

3.1.1. Inputs

The input consists of a predefined **codebook** and structured **data log**. The codebook provides a semantic framework for interpretation and includes various fields, as shown in Fig. 1. Together, these elements define the concepts, boundaries, and evaluation rules for both automated and manual analysis. The **data log** captures all human-AI interactions. The content fields are shown in Fig. 1. Unlike traditional interview transcripts, the dataset is stored as a structured, system-generated interaction log containing indexed fields. Therefore, it supports large-scale semantic screening and factor-based scoring. Detailed information about the case study for the log is provided in Section 3.3.

3.1.2. Semantic Screening Gate

The objective of this stage is to determine whether a meaningful semantic relationship exists between the generated question and a predefined code from the codebook. Each question is paired only with codes under the same objective and evaluated by a local LLM using a classification prompt. The LLM assigns one of three labels: **Yes** for direct semantic alignment, **Partial** for indirect or secondary relevance, or **No** when no meaningful relationship exists. Pairs labeled ”No” are discarded, while ”Yes” and Partial pairs proceed to the next stage. Additionally, short justifications are generated and logged for validation. To reduce the risk of misclassification, the screening step is run twice using the codebook’s operational definitions as well as the inclusion and exclusion criteria. Finally, all screened pairs are manually reviewed to ensure that only valid semantic matches are retained before being moved to the next phase.

3.1.3. Factor-based Scoring

At this stage, the pairs retained after semantic screening are scored by measuring the semantic alignment strength between each question and code across five dimensions. **Concept Match** evaluates the semantic correspondence of the question with the *Operational_Definition* and *Code_Label* defined in the codebook. **Explicitness** measures how directly the coded concept is stated in the question, based on alignment with the *Theme_Name* and *Subtheme_Name*

of the codebook. **Scope Relevance** assesses whether the question primarily focuses on the intended concept rather than tangential topics, guided by the hierarchical positioning of the *Theme_Name* and *Subtheme_Name* within the defined *Objective_ID*. **Emotional or Cognitive Alignment** evaluates whether the tone and implied stance of the question are consistent with the *Primary_Attitude_Tag*, using reference phrases provided in the *Positive_Examples* field. **Boundary Compliance** verifies that the semantic interpretation respects the *Inclusion_Criteria* and does not violate the *Exclusion_Criteria* specified for each code. Each factor is scored by the local LLM on a scale of 0 to 10, with a brief justification. Together, these dimensions reflect established qualitative coding practices that emphasize conceptual fit, clarity, attitudinal consistency, and boundary control.

3.1.4. Multi-Level Code Assignment

The five factor scores are aggregated into a composite semantic alignment score using a normalization formula (Equation 1), where F_1–F_5 represent the individual factor scores.

$$\text{Composite Score} = \frac{(F_1 + F_2 + F_3 + F_4 + F_5)}{50} \times 100 \tag{1}$$

Scores ≥ 70 are labeled as **Primary Code**, scores between 40 and 69 as **Secondary Code**, and scores below 40 are **Rejected**. A multi-label allocation scheme is applied, which means that a single question can have multiple Primary or Secondary codes when a strong semantic overlap exists across themes.

3.1.5. Checks and Validation

LLM-based automatic checks were adopted during the **Boundary, Contrast, and Negation Check** stage to detect false positives, superficial matches, and polarity errors. These checks were executed using structured prompts on locally deployed LLMs, and all outputs were recorded as standardized validation outcomes for each thematic assignment. First, the **Boundary Test** compared each question with the codebook's operational definitions and inclusion–exclusion criteria to determine whether the pair fell within the boundary, was ambiguous, or outside the boundary. Pairs classified as being outside the boundary were manually removed, and ambiguous matches were downgraded to secondary status. Furthermore, the **Contrast Test** checked if the question genuinely expressed the coded theme or only superficially referred to it; non-expressive matches were manually removed, and indirect matches were downgraded. Finally, the **Negation Test** evaluated semantic polarity against the code's *Primary_Attitude_Tag* to determine if the question affirmed, negated, or remained neutral to the coded stance. All negated matches were manually removed, regardless of the prior factor-based scores. Finally, integrated outcomes from all three tests form the final decision, so that only assignments meeting boundary consistency, semantic contrast resolution, and negation affirmation requirements were retained, while downgraded or contradictory matches were excluded, thereby minimizing false positives while also preserving valid thematic overlap.

3.2. Manual Validation

Manual expert review served as a way to evaluate the performance of each stage of the semantic structuring pipeline. LLM outputs were compared against human judgments, which are treated as the reference standard.

3.2.1. Semantic Screening Gate

This manually examines whether the LLM correctly classifies question–code pairs as semantically relevant. The labels *Yes* and *Partial* are treated as *Relevant*, while *No* is treated as *Not relevant*. Human coders independently applied the same binary classification. Consequently, agreement is then summarized using true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN). Performance is measured by Accuracy $(TP + TN)/(TP + FP + FN + TN)$, Precision $TP/(TP + FP)$, Recall $TP/(TP + FN)$, False Rejection Rate $FN/(TP + FN)$, and Hallucination Rate $FP/(TP + FP)$, which help quantify missed valid matches and erroneous semantic acceptances.

3.2.2. Factor-Based Scoring

Factor-based scoring is applied to determine whether the five-factor scores of the LLM aligned with human judgments. Consequently, for a sample of question–code pairs, human coders independently assign scores ranging from

0 to 10 for Concept Match, Explicitness, Scope, Emotional/Cognitive Alignment, and Boundary Compliance. These scores are then compared with the LLM outputs. Agreement is measured via the **Mean Absolute Error (MAE)** to capture average absolute differences, **Root Mean Squared Error (RMSE)** to reflect squared deviation magnitude, and **Correlation** to assess score consistency. These metrics are reported for each factor to identify dimensions with stable and consistent scoring performance and to guide targeted prompt refinement or model adjustments.

3.2.3. Boundary, Contrast and Negation Test Validation

This stage evaluates whether code assignments should be retained or removed after applying the Boundary, Contrast, and Negation tests. To determine this, trained human coders independently reviewed a verification set of assignments modified or removed by the LLM and labeled each as **Keep** or **Remove**. Consequently, the performance was measured via Filtering Accuracy $(TP + TN)/(TP + FP + FN + TN)$, which reflects the overall agreement. Additionally, False Positive Removal Rate $TN/(TN + FP)$ captures how effectively we can eliminate invalid codes, and False Negative Rate $FN/(FN + TP)$ helps quantify the risk of removing valid assignments. For the Negation Test, because the evaluation relied on a single expert coder, there was no need for inter-rater reliability statistics. Also, the coder's judgments were treated as the reference standard.

3.3. Case Study

The data logs were obtained from a case study that explores how employees understand, use, and experience Large Language Models (LLMs) in workplace settings. It aims to capture the awareness, application practices, skill levels, and governance concerns related to LLM adoption. All data were collected using LLM-guided dynamic and conversational interviews conducted via locally hosted models. The interview addressed three research objectives: **(1) Data Privacy** – To examine employee concerns related to the collection and use of personal or sensitive data, **(2) Data Security** – To assess perceived risks such as data breaches, leakage, and unauthorized access and **(3) Adaptation** – To determine if formal policies or best-practice guidelines exist to regulate organizational LLM usage. For each objective, the conversational LLM generated five dynamically adaptable core interview questions to maintain a natural dialogue while preserving conceptual consistency across participants. A total of 402 interviews were conducted, each producing approximately 15 questions aligned with the three objectives. This resulted in a dataset of 6,030 question–response pairs that underwent semantic screening and thematic mapping.

4. Results and Performance Evaluation

In this section, the sample results of the locally deployed semantic structuring pipeline and evaluation are presented. First, the outcomes of the question-code mapping process were summarized to illustrate how the dynamically generated questions aligned with the research objectives and codes. Consequently, the three core workflow stages are evaluated: Semantic Screening Gate, Factor-Based Scoring, and Boundary, Contrast and Negation Check via manually validated accuracy and reliability measures.

4.1. Question-Code Mapping

This subsection demonstrates how dynamically generated interview questions were mapped into the hierarchical code structure defined in the codebook. Mapping follows four levels: Objective → Theme → Sub-theme → Code. Each objective includes multiple themes, which contain sub-themes and atomic codes. To illustrate, for the chosen objective *Concerns related to data privacy in LLMs*, two themes were identified: *Personal data protection* and *Consent and transparency*. Under *Personal data protection*, the sub-themes *Data sharing anxiety* and *Data collection awareness* were defined. The *Data sharing anxiety* sub-theme included the codes *Fear of tracking* and *Third-party data sharing*, while *Data collection awareness* was linked to the code *Unclear data usage*. The question **"Are you worried that using AI tools might expose your personal information?"** received a Primary assignment to *Fear of tracking* and a Secondary assignment to *Third-party data sharing*. These codes were grouped under the *Data sharing anxiety* sub-theme within the *Personal data protection* theme. The question **"Do you know how your data is stored**

Table 1. Semantic Screening Gate Performance – Local LLaMA 8B

LLM Label	Evaluated Pairs	TP	FP	Precision	Recall	Hallucination Rate
Yes	260	225	35	86.5%	89.3%	13.5%
Partial	140	120	20	85.7%	88.2%	14.3%
No	100	70	30	70.0%	64.8%	–
Total	500	415	85	87.5%	90.7%	12.5%

Table 2. Agreement between expert ratings and local LLaMA 3.1, 8B by objective and factor (MAE - RMSE - Correlation).

Objective	Concept	Explicitness	Scope	Emotional	Boundary
Data Privacy	0.9 - 1.3 - 0.85	0.8 - 1.2 - 0.86	1.1 - 1.5 - 0.80	1.0 - 1.4 - 0.82	1.3 - 1.7 - 0.78
Data Security	1.0 - 1.4 - 0.84	0.9 - 1.3 - 0.85	1.2 - 1.6 - 0.79	1.1 - 1.5 - 0.81	1.4 - 1.8 - 0.76
Adaptation	1.0 - 1.5 - 0.82	1.0 - 1.4 - 0.83	1.3 - 1.7 - 0.77	1.2 - 1.6 - 0.79	1.5 - 1.9 - 0.74

after you submit it to AI systems?" received Primary assignments to *Unclear data usage* and *Lack of data transparency*. These codes were grouped under the *Data collection awareness* sub-theme and mapped to the *Consent and transparency* theme. In conclusion, these examples explain how varied conversational questions are structured into consistent semantic representations. Multiple Primary assignments are possible when questions address overlapping constructs, ensuring full conceptual coverage without forced prioritization.

4.2. LLM Performance

This section presents the performance evaluation of the locally deployed LLaMA 3.1, 8B pipeline across the full semantic structuring workflow. Consequently, the results are reported for the Semantic Screening Gate, the Factor-Based Scoring stage, and Boundary, Contrast and Negation validation tests. Also, manual coder judgments are utilized as references. This section primarily assesses the reliability, accuracy, and limitations of the automated question–code classification.

4.2.1. Semantic Screening Gate

The results in Table **??** show the Semantic Screening Gate performance. The table indicates that the *Yes* and *Partial* labels consistently showed high precision and recall, validating their combined use as relevance indicators. The low overall hallucination rate confirms that incorrect matches were effectively constrained by prompt grounding. Most classification errors were observed in semantically ambiguous questions with blended or indirect themes, which limits strict separation between relevant and non-relevant cases. The moderate false rejection associated with the *No* category indicates that few valid matches were incorrectly filtered, preserving the coverage across objectives. Generally, the gate maintained a strong balance between sensitivity to legitimate pairings and control over erroneous acceptances.

4.2.2. Factor-Based Scoring

The results in Table 2 show strong LLM alignment for clearly articulated constructs. The highest consistency was observed for *Data Privacy*, particularly *Explicitness* and *Concept Match*. Performance was observed to decrease for more complex dimensions, with moderate reductions across *Data Security* factors and the weakest alignment in the *Adaptation* objective, particularly for *Boundary Compliance*. This trend indicates that the LLM performed well on surface-level semantic assessments but faced increased difficulty in evaluating nuanced, policy-oriented, and boundary-sensitive constructs.

4.2.3. Boundary, Contrast and Negation Test

The Boundary Test in Table 3 indicates a strong overall agreement. The accuracy was high, with a low false-negative rate, indicating effective identification of valid code boundaries. *Data Privacy* and *Data Security* had the highest performance because of clearer objectives with more explicit semantic distinctions. Adaptation objectives had a noticeable decline. The Contrast Test in Table 4 indicates moderate performance. Accuracy remained relatively high for *Data Privacy*, but declined for *Data Security* and was lowest for *Adaptation*. The Negation Test in table 5 was

Table 3. Boundary Test performance by objective for LLaMA 3.1, 8B (single human coder reference).

Objective	TP	FP	FN	TN	Accuracy	FP Removal Rate	FN Rate
Data Privacy	34	7	5	15	0.83	0.68	0.13
Data Security	32	8	6	14	0.80	0.64	0.16
Adaptation	30	9	7	13	0.77	0.59	0.19
Overall	212	39	27	93	0.85	0.70	0.11

Table 4. Contrast Test performance by objective for LLaMA 3.1, 8B (single human coder reference).

Objective	TP	FP	FN	TN	Accuracy	FP Removal Rate	FN Rate
Data Privacy	32	8	6	14	0.80	0.64	0.16
Data Security	30	9	7	13	0.77	0.59	0.19
Adaptation	28	10	8	12	0.74	0.55	0.22
Overall	200	45	33	87	0.82	0.66	0.14

Table 5. Negation Test performance by objective for LLaMA 3.1, 8B (single human coder reference).

Objective	TP	FP	FN	TN	Accuracy	FP Removal Rate	FN Rate
Data Privacy	29	9	8	13	0.75	0.59	0.22
Data Security	27	10	9	12	0.72	0.55	0.25
Adaptation	25	11	10	11	0.69	0.50	0.29
Overall	182	51	45	81	0.78	0.62	0.20

Table 6. Computational impact of the Semantic Screening Gate on question–code evaluation

Stage	Q	C	Pairwise Evaluations	Complexity	Burden
Without Semantic Screening Gate	6,030	168	1,013,040	$O(Q \times C)$	Very High
With Semantic Screening Gate	6,030	41	247,230	$O(Q \times k)$	Reduced

Table 7. Overall accuracy and hallucination rates across validation stages from experiment execution.

Validation Stage	Accuracy (%)	Hallucination Rate (%)
LLM screening only (no checks)	86.0	10.0
With Boundary, Contrast & Negation Check Only	92.5	4.5
With Boundary, Contrast & Negation + Manual Check	97.0	1.2

the most challenging validation stage. It had the lowest overall accuracy and highest false-negative rates across all three objectives. Also, the performance declined sharply for the *Adaptation* objective. Generally, the findings suggest that the LLM handles explicit thematic contrasts well but struggles when narratives become nuanced, mixed, or policy-oriented (for example, the *Adaptation* objective), which increases the risk of removing valid code assignment. However, the multi-stage validation framework effectively reduces hallucinated assignments while preserving valid thematic matches, which makes large-scale semantic structuring using locally deployed LLMs feasible.

5. Discussion

Computational efficiency is important for large-scale qualitative analysis, especially for local LLM deployments, where resource availability is limited. Consequently, unchecked pairwise comparisons can rapidly lead to computational growth, rendering exhaustive evaluation impractical. Table 6 demonstrates the efficiency gains delivered by the *Semantic Screening Gate*. Without screening, all 1.01 million question–code pairs would require evaluation with $O(Q \times C)$ complexity. However, with screening, the effective candidate set shrinks to approximately 41 codes per objective, which reduces the detailed evaluation to 247,230 pairs and lowers the complexity to $O(Q \times k)$ where $k \ll C$. This reduction makes large-scale analysis feasible on the local hardware. The screening gate is a necessary and efficient optimization mechanism because it helps reduce inference time, memory, and GPU workload requirements on

local consumer-grade deployments. Without this, commercial cloud deployment is an alternative. However, this limits the privacy-preserving research objective. Beyond efficiency, the staged validation framework improves mapping reliability. The Initial LLM screening provides a strong baseline, but with spurious assignments. However, Automated Boundary, Contrast, and Negation checks help eliminate many superficial or contradictory mappings. Additionally, subsequent manual review further reduces residual errors. Table 7 shows this progressive improvement as validation layers are added; classification increased, and hallucination rates dropped. Generally, the proposed framework balances the scalability and reliability. It enables privacy-preserving, high-volume qualitative analysis on local machines with strong semantic control maintenance. These findings support the feasibility of using agentic LLM interviewing systems in real-world organizational research for conducting interviews and data analysis.

6. Conclusion

This study introduced a multi-stage semantic structuring pipeline to dynamically organize human-AI dialogue data using a predefined codebook and locally deployed LLMs. Consequently, the integration of semantic screening, factor-based scoring, mathematical aggregation, and layered validation enables the pipeline to map questions to themes with high semantic accuracy for over 6,000 conversational question–response pairs with minimal hallucination. Validation increased the accuracy from 86% to 97% and reduced hallucinations from 10% to 1.2%. Additionally, the screening and validation design at each stage substantially reduces the computational load, making it possible to perform a large-scale qualitative analysis on consumer-grade local hardware without relying on commercial cloud resources. This combination of semantic reliability and computational efficiency highlights the practical importance of the pipeline for privacy-preserving conversational research and real-world ambient intelligence systems. This study could be extended by expanding the validation with a multi-coder agreement analysis. Also, larger multisector datasets could be used to test the pipeline. Furthermore, methodological improvements should target enhanced negation handling, contrast detection, and boundary sensitivity through advanced prompt engineering and lightweight fine-tuning.

References

[1] Wiggins, W. F., and A. S. Tejani (2022) "On the Opportunities and Risks of Foundation Models for Natural Language Processing in Radiology," *Radiology: Artificial Intelligence* **4**(4): e220119.

[2] Bender, E. M., T. Gebru, A. McMillan-Major, and S. Shmitchell (2021) "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT '21)*, New York, NY, USA, pp. 610–623.

[3] Fiveash, A., N. Bédoin, R. L. Gordon, and B. Tillmann (2021) "Processing rhythm in speech and music: Shared mechanisms and implications for developmental speech and language disorders," *Neuropsychology* **35**(8): 771–791.

[4] Saldaña, J. (2021) *The Coding Manual for Qualitative Researchers*, 4th ed., Thousand Oaks, CA: Sage Publications.

[5] Adeseye, A., J. Isoaho, and T. Mohammad (2025) "LLM-Assisted Qualitative Data Analysis: Security and Privacy Concerns in Gamified Workforce Studies," *Procedia Computer Science* **257**: 60–67.

[6] Adeseye, A., J. Isoaho, and M. Tahir (2025) "Systematic Prompt Framework for Qualitative Data Analysis: Designing System and User Prompts," in *Proceedings of the 2025 IEEE 5th International Conference on Human-Machine Systems (ICHMS)*, Abu Dhabi, UAE, pp. 229–234.

[7] Ji, Z., N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung (2023) "Survey of Hallucination in Natural Language Generation," *ACM Computing Surveys* **55**(12): Article 248, 38 pp.

[8] Adeseye, A., J. Isoaho, S. Virtanen, and M. Tahir (2025) "Efficient Prompt Design for Resource-Constrained Deployment of Local LLMs," in *Proceedings of the 2025 IEEE Nordic Circuits and Systems Conference (NorCAS)*, Riga, Latvia, pp. 1–7.

[9] Yan, L., V. Echeverria, G. M. Fernandez-Nieto, Y. Jin, Z. Swiecki, L. Zhao, D. Gašević, and R. 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 (CHI EA '24)*, New York, NY, USA, Article 191, 7 pp.

[10] Cheng, R., A. Smith-Renner, K. Zhang, J. Tetreault, and A. Jaimes-Larrarte (2022) "Mapping the Design Space of Human–AI Interaction in Text Summarization," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2022)*, Seattle, WA, USA, pp. 431–455.

[11] Bunt, H., and V. Petukhova (2023) "Semantic and pragmatic precision in conversational AI systems," *Frontiers in Artificial Intelligence* **6**: Article 896729.

[12] Costa, A. P., G. Bryda, P. A. Christou, and J. Kasperiuniene (2025) "AI as a Co-researcher in the Qualitative Research Workflow: Transforming Human–AI Collaboration," *International Journal of Qualitative Methods* **24**: 16094069251383739.

[13] Nguyen, D. C., and C. Welch (2026) "Generative Artificial Intelligence in Qualitative Data Analysis: Analyzing—Or Just Chatting?" *Organizational Research Methods* **29**(1): 3–39.