From Tacit Knowledge to Structured Documents: A Framework for Knowledge Elicitation

Sylvain Roudiere and Bianca Lento
LR Technologies, Toulouse, France

sroudiere@lrtechnologies.fr
blento@lrtechnologies.fr

Abstract: Recent advances in Large Language Model (LLM) "agent" systems have moved language models beyond single-turn generation toward goal-directed workflows that can plan, ask clarifying questions, and iteratively refine outputs. In parallel, retrieval-augmented generation (RAG) has become a practical way to ground these agents in enterprise knowledge—enabling models to leverage internal documentation and policies without expensive and recurring retraining. However, RAG is only as strong as the underlying corpus: when key information is missing, outdated, or fragmented, retrieval cannot fill the gaps. In many organizations, assembling high-quality, up-to-date documents remains a persistent bottleneck, limiting both the reliability of downstream applications and the speed at which knowledge can be operationalized. To address this, we introduce the Assistant-Scribe-Knowledge Checker (ASK) framework for generating structured specification documents through guided interviews. ASK decomposes the end-to-end workflow into three specialized LLM agents: an Assistant that asks questions to the user and steers the interview through adaptive follow-ups; a Scribe that continuously summarizes what has been said and records it into a schema-constrained specification document with explicit content requirements; and a Knowledge Checker that evaluates the evolving document against those specifications, detects gaps or weakly supported entries, and advises the Assistant on the next best questions to ask to reach the desired completeness and quality. This separation supports seamless document creation while improving quality control over content, structure, and coverage. By distributing responsibilities across three lightweight, role-specific agents, ASK reduces reliance on a single large model and enables modular scaling across teams and domains. We evaluated the ASK framework in a consulting-firm setting where consultants are required to produce project "return-of-experience" documents (successes, failures, challenges, and solutions) to capture reusable knowledge. Compared to documents authored manually, ASK-guided interviews consistently produced more complete, better-structured, and higher-quality specifications with less variability across authors. Beyond measurable quality gains, consultants also reported a clear preference for the interview-based workflow, citing lower effort and a more natural way to articulate tacit project knowledge.

Keywords: Knowledge management, Agentic AI, Large language models, Generative AI, Knowledge elicitation

1. Introduction

Recent advancements in the capabilities of Large Language Models (LLMs) have renewed interest in AI-driven knowledge management. As these models demonstrate increased proficiency in interpreting unstructured information, synthesizing dispersed evidence, and generating natural language responses, they provide a robust foundation for systems that help organizations manage and mobilize internal knowledge. This is particularly critical in modern companies, where the scale and speed of business operations frequently surpass the capacity of unaided human effort (Microsoft WorkLab 2025).

Employees are required to navigate increasing volumes of information across multiple platforms while maintaining productivity and decision quality, making agentic AI a compelling solution for augmenting organizational processes. Concurrently, the demand for knowledge management technologies is rising (Grand View Research 2025; Singla et al. 2025), reflecting a widespread recognition that knowledge accessibility constitutes a strategic organizational capability. However, organizational knowledge often remains hard to access because it is dispersed, siloed, or never explicitly documented in the first place (Atlassian 2025; Malgard et al. 2024). Much of the most valuable knowledge in a company exists tacitly in the experience of employees, rather than in structured repositories.

Consequently, effective AI-driven knowledge management requires not only better retrieval, but also better elicitation of undocumented expertise. A knowledge elicitation agent addresses this gap by helping transform tacit, fragmented knowledge into explicit and operational knowledge resources.

2. State of the Art

Capturing the knowledge held by employees - particularly tacit knowledge, which Polanyi (1966) defined as knowledge that individuals possess but cannot easily articulate - has long been recognized as a critical challenge for organizations. Nonaka and Takeuchi (1995) formalized this challenge through their SECI model, describing how tacit knowledge must be externalized into explicit form before it can be shared and reused.

Traditional elicitation methods, such as structured interviews, think-aloud protocols, and observation sessions, remain resource-intensive and difficult to scale. A systematic review by Rosario and Amaral (2021) identified social, technical, and organizational barriers that hinder elicitation efforts, while Kerrigan et al. (2021) found that most processes remain ad hoc, poorly documented, and lack methodological rigor. These findings highlight the need for more systematic and scalable approaches to capture knowledge.

Conversational agents have emerged as a promising avenue for addressing these scalability issues. In the space domain, Mihaylov et al. (2022) developed SCARLET, a conversational agent designed to elicit expert-level tacit knowledge from mission personnel for structured retrieval. SCARLET relied on rule-based dialogue management and a dedicated question generation module, demonstrating the potential of automated knowledge capture but also exposing the limitations of non-generative architectures in handling complex, open-ended conversations. In the manufacturing domain, Freire et al. (2023) introduced CLAICA, a continuously learning cognitive assistant that captures knowledge from experienced production-line workers through conversational interaction, formalizes it, and shares it with novices. A user study with 83 participants yielded design guidelines regarding the effects of prior training, context expertise, and interaction modality on user experience.

These systems demonstrated that conversational interfaces can effectively support knowledge exchange, but they were built on pre-LLM natural language understanding pipelines, which limited their ability to handle diverse topics and generate contextually rich follow-up questions.

The advent of LLMs has fundamentally shifted the landscape. Studies suggest that LLMs can effectively support tacit knowledge elicitation from operators through natural language interaction, yet challenges around user acceptance and output quality persist (Bubeck et al. 2023; Chen et al. 2025; Dwivedi et al. 2023; Lee, Jung, and Baek 2024). In Kuks et al. (2025), a ChatGPT-5-based chatbot conducted semi-structured expert interviews and produced structured summaries for a knowledge database. Experts rated the interviews positively, but the summaries showed gaps in completeness and occasional hallucinations, highlighting the gap between conversational quality and reliable output. Korn et al. (2025) introduced LLMREI, a GPT-4o-powered chatbot for requirements elicitation interviews using two prompting strategies: minimal zero-shot and iteratively refined least-to-most prompts. Across 33 simulated interviews, the refined prompt matched trained human interviewers in error rates and captured approximately 70% of intended requirements, though limitations included missing non-verbal cues, overly broad questioning, and challenges with long-term coherence. Similarly, Görer and Aydemir (2023) explored LLM-generated interview scripts, finding that task decomposition improved coherence but complex conversation graphs remained challenging.

While these studies demonstrate the potential of LLM-driven automated interviews, they focus primarily on software requirements engineering and produce outputs as requirement lists or raw transcripts. None of them address the broader challenge of corporate knowledge elicitation nor the generation of structured documents conforming to predefined specifications. Our work bridges this gap by proposing a framework in which an LLM autonomously interviews employees according to a document specification, progressively building a structured document as the conversation unfolds. To our knowledge, this combination of specification-driven document generation and automated conversational elicitation is a novel contribution at the intersection of knowledge management, conversational AI, and LLM-based document automation.

3. Material and Methods

3.1 System Requirements

We designed this framework around a set of core objectives and constraints. First, we aimed to develop a system that can run on reasonably sized machines, enabling companies to deploy it locally with limited infrastructure investment while maintaining full control over their internal data and avoiding reliance on remote LLMs. Second, we sought to ensure compatibility with small-to-medium-sized LLMs in order to improve scalability and accessibility. Third, we wanted the framework to remain agnostic to document structures and context, so that it could be adapted to a wide range of use cases. Finally, we aimed to provide a system capable of seamless interaction throughout the knowledge acquisition process.

The design of the framework was inspired by interview-based knowledge gathering practices commonly used in companies, where an interviewer elicits information from an employee, takes notes, and continuously adapts subsequent questions according to the information collected during the exchange.

3.2 Architecture

We introduce ASK, a novel architecture for document elicitation shown in Figure 1. This architecture is based on three specialized stateful LLM agents: the Assistant, the Scribe and the Knowledge Checker. These agents interact by modifying a shared global state, which allows the document to be progressively constructed as the user interview is conducted. The document drafting process is guided by drafting instructions, that define the expected content and specify requirements for each section and subsection.

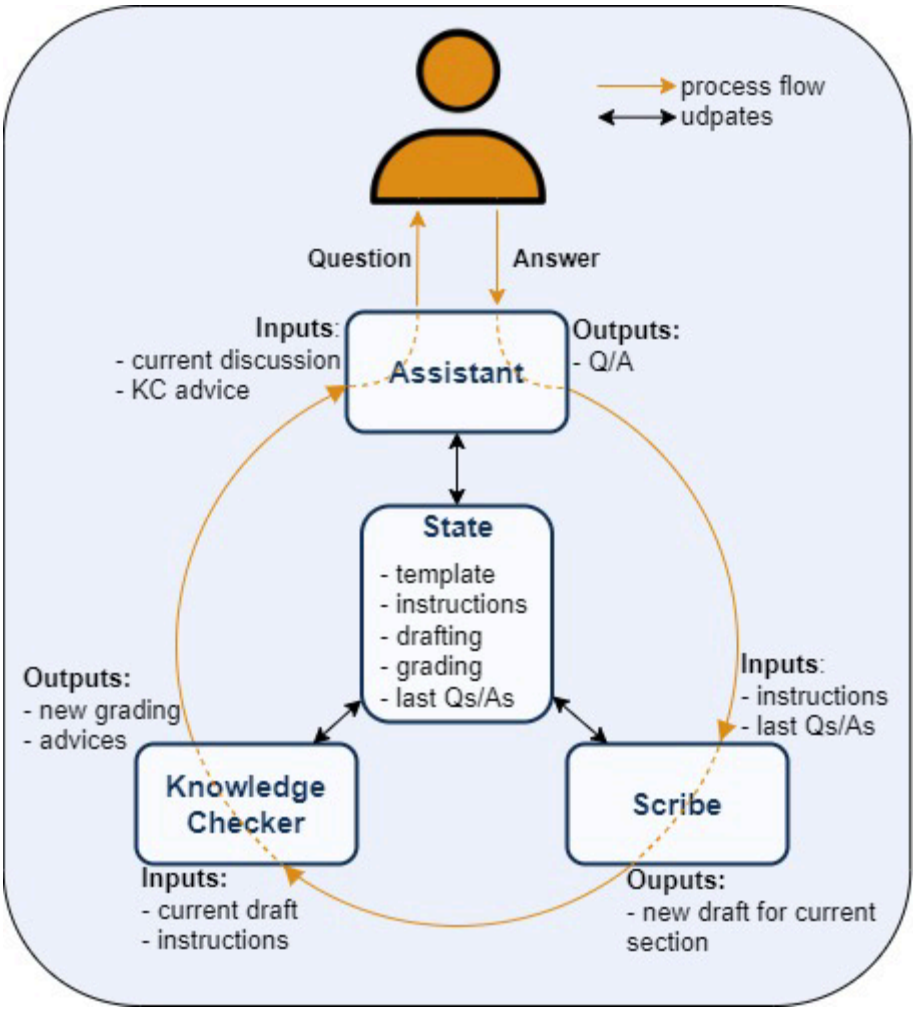

Figure 1: ASK framework for knowledge elicitation

3.2.1 Assistant

The Assistant is the agent performing the interview. It is the only agent the user interacts with. Its role is to pose questions to the user in accordance with the guidance provided by the Knowledge Checker. Its inputs are the current discussion it's having with the user, as well as the advice given by the Knowledge Checker. Once the user replies with an answer, the Scribe starts writing.

3.2.2 Scribe

The Scribe is the agent responsible for drafting the document. It takes as input the latest question-answer exchange between the Assistant and the user, along with the drafting instructions for the current section. The Scribe then produces an updated version of the section draft, incorporating the new information obtained from the exchange. By limiting its focus to the current section and its associated requirements, the Scribe reduces the likelihood of hallucinations or unnecessary modifications to unrelated sections.

3.2.3 Knowledge checker

The Knowledge Checker (KC) is designed to emulate the role of a human interviewer by guiding the interview to elicit information that satisfies the document requirements. It considers the context previously provided by

the user, ensuring that questions remain relevant and dynamic. The KC takes as input the current section draft and its associated requirements. For each requirement, it assigns a score from 0 to 5 according to the following scale: 0 - not met at all, 1 - not met, 2 - almost met, 3 - barely met, 4 - met, and 5 - met beyond expectations. This score is then used to select a requirement and its associated subsection, and make the KC generate an advice for the Assistant to stir the discussion towards fulfilling the selected requirement.

3.2.4 Drafting instructions

The drafting instructions have the structure described in Figure 2.

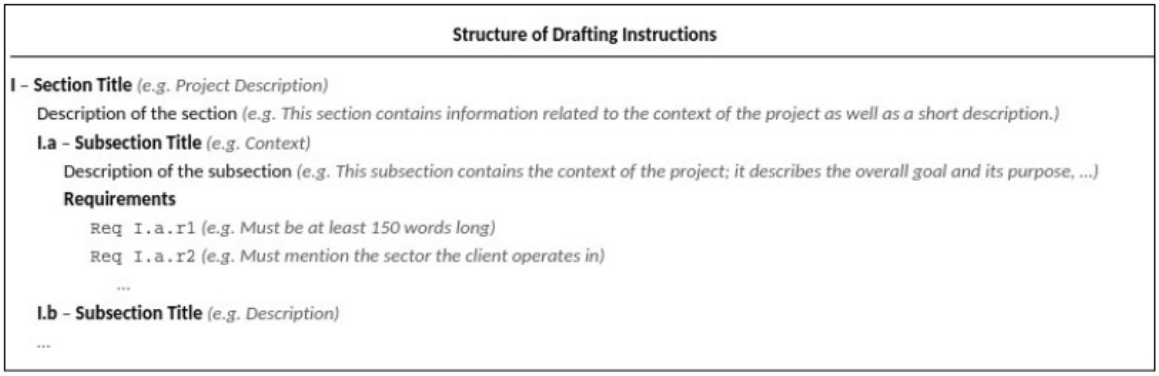

Figure 2: Drafting instructions structure

The Scribe is provided with the complete drafting instructions for the current section, including section and subsection identifiers used to structure the output and ensure that updates are applied to the correct subsection. Requirement identifiers are not provided to the Scribe, as they are not necessary for drafting and may be misinterpreted as subsection identifiers.

In contrast, the KC has access to requirement identifiers, which are used to associate each requirement with its corresponding evaluation score. Additionally, a word count is dynamically computed for each subsection to assist the agents in satisfying length-related requirements.

For each section of the document, the process goes iteratively through the following steps:

- The Assistant asks the user a question and receives a response.
- The Scribe then follows the available drafting instructions to update the draft of the current section using the information obtained from the latest question-answer exchange.
- The KC evaluates the newly drafted version of the current section to assess its content quality. It assigns a score ranging from 0 (requirement not met) to 5 (requirement met beyond expectations) for each requirement associated with the section under consideration.
- If all requirements receive a score of 4 or 5, the section is considered validated, and the drafting process proceeds to the next section (or terminates if the last section has been completed). Otherwise, a requirement with a score of 3 or lower is selected, and the KC generates guidance for the Assistant to steer the next question toward addressing the unmet requirement.

3.2.5 LLM model

During development and experimentations, the model used was Gemma 3 27B. This model was chosen because of its lightweight and relatively good performance on benchmark related to summarization at the time. It is representative of the type of model a company could run with relatively low investment as it can run, unquantized, on a single high-end GPU.

3.3 Mission Reporting Document

To assess the ASK framework, we conducted a pilot study involving a subset of employees within the company. As part of data collection for the knowledge management system, each employee must complete a Mission Reporting Document (MRD). This document requires administrative information about the employee, details of their mission, and information about the current client. The remainder of the document, however, is left in a relatively free format. Its primary purpose is to collect information on the following aspects:

- Context of the mission

- Objectives (success criteria, who will benefit from the work done)
- Role of the employee, their tasks, their contribution, their deliverables
- Technological context of the mission (technologies used, algorithms, technological components, …)
- The challenges, obstacles or difficulties encountered during the mission or linked to the mission.
- The mission evolutions (if the scope/planning of the mission changed, how and why)
- The results of the mission (ideally with metrics)
- The next steps of the mission

The document is divided into 4 sections: Administrative information (name, project name, job), Context and Objectives, Consultant role and tasks, Results and perspectives. For each section of the document, employees are provided with example questions intended to guide the writing process. Yet, as a consulting company, our employees operate across a wide range of clients, contexts, technologies, methodologies, and skill sets, which introduces significant variability in document content. As a result, assessing the quality of these documents is a complex task. Currently, this evaluation is performed by a senior employee with extensive functional expertise, who reviews the MRDs and provides feedback via email or schedules meetings when critical or essential information is missing. Employees are required to complete an MRD every three months. After one year of data collection, only 25% of the expected 1,200 documents had been submitted, and among these, 33% required additional feedback.

3.4 Pilot Study

We evaluated the use of the ASK architecture for MRD drafting by developing an application with which employees could interact. The user interface (UI) is illustrated in Figure 3. The app has been built in Python with the Streamlit library as its ease of implementation makes it perfect for proof-of-concept development.

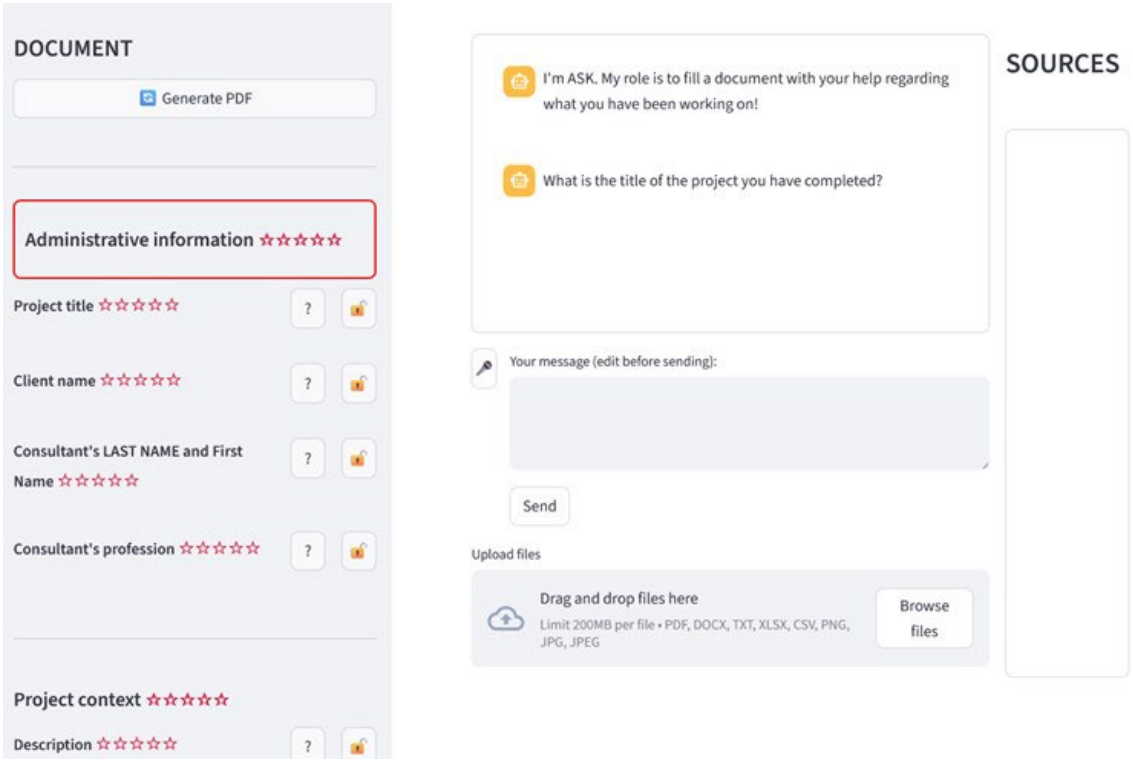

Figure 3: Streamlit app built to interact with ASK

The UI enables interaction with the ASK agents and provides several functionalities. Users can input responses either by typing or by using a local, real-time speech-to-text feature, allowing them to dictate their answers. The transcription, generated using the *faster-whisper* library with a locally deployed large Whisper model, can be edited before submission. The interface also allows users to view the current draft of all document sections, upload files for automatic content extraction and completion, and submit images to be included in the final document.

To mitigate common limitations of LLMs, additional features were introduced: a lock button enables users to "lock" a subsection, meaning mark it as final, preventing further modifications by the agent; and a "?" icon allows users to inspect subsection requirements along with their current evaluation scores.

Users were instructed to interact with the ASK agents until the system indicated that the drafting process was complete. Prior to this interaction, they were provided with a brief overview of the interface and its functionalities.

After the completion of each section, a validation page is presented. On this page, users are asked to review the generated content and make corrections if necessary. They are also asked to rate from 1 to 10 their overall satisfaction with the section and provide open-ended feedback. This feedback is used to identify friction points in the human-AI interaction process.

4. Results

A total of 12 participants were involved in the study, covering a range of job roles (five developers, one QA engineer, four industrial engineers, and two project managers). The experimentation was conducted using the Gemma 3 27B model, deployed on a system equipped with two NVIDIA GeForce RTX 4090 GPUs. As the first section of the document consists solely of administrative information (name of the consultant, client, role), it was excluded from the statistical analysis, as it does not reflect active interaction with the ASK agents.

Sessions lasted on average 94.3 minutes, representing a substantial reduction from the 4-5 hours previously required for manual completion of the MRD by consultants.

Table 1: Distribution of time spent in each ASK state during experimentations

State	Avg total time per session (min)	Avg time per occurrence (min)
Input phase	71.1	5.8
of which audio recording	9.1	1.0
Knowledge Checker	8.9	0.5
Scribe	3.3	0.3
Assistant	1.1	0.1
Feedback Review	9.2	2.3
Final Feedback	2.5	2.5

Table 1 shows the distribution of time spent in each state of the ASK architecture. As expected, the input phase represents most of the time spent on the app even with the availability of audio transcription. On average, users waited 54.4 seconds between input phases while the agents completed their processing: 16.2 seconds for the Scribe, 32.7 seconds for the KC and 5.5 seconds for the Assistant. On average, 12.3 user-agents exchanges were required to complete the MRD: the agent asked an average of 29.6 questions, as it usually posed two or three questions within the same message.

Table 2 shows the average time spent for each of the MRD sections with their corresponding satisfaction rating from the user. A clear correlation was observed between the time spent on a section and the corresponding user satisfaction rating. Section 3 required the most time to complete, despite also being the most frequently locked section, with seven out of twelve participants fully locking it to force the ASK agent to proceed to the next section.

Table 2: Distribution of time spent in each MRD section during experimentations

Section	Time spent of total (%)	Satisfaction Rating (/10)
2 - Context & Objectives	35.8	7.1
3 - Task performed	45.3	5.4
4 - Results and perspectives	18.9	7.9

Several key insights emerged from the open-ended feedback collected during the study:

- Users were generally satisfied with the quality of the agent's writing but noted that proofreading remains necessary as the agent occasionally produced hallucinated content.

- Users found the questions posed by the agent to be highly relevant, although sometimes it became repetitive by asking similar questions.
- Processing times were perceived as slightly long, indicating potential for optimization.

The quality of MRDs produced by ASK was evaluated through systematic comparison with manually assessed non-ASK MRDs. A corpus of 191 documents was labelled across three quality tiers — low, medium, and high — by a senior domain expert responsible for data collection, based on criteria of content quality and completeness. A set of quantitative metrics was subsequently extracted from both the manually classified documents and the ASK-generated documents. These metrics comprised word count and six content quality dimensions assessed by an LLM-based judge (Gemma 3 27B): Grammar, Structure, Vocabulary, Clarity, Completeness, and an overall Score, each rated on a ten-point scale.

As illustrated in Figure 4, a clear positive correlation is observed between the quality labels assigned by the domain expert and the scores produced by the LLM judge. Specifically, low-quality MRDs received a mean overall grade of 5.33, medium-quality MRDs a mean of 5.94, and high-quality MRDs a mean of 6.47. This monotonic relationship holds consistently across all individual metrics, including word count, where low-quality documents contain on average three times fewer words than their high-quality counterparts.

The 12 MRDs generated by ASK achieved a mean LLM grade of 6.84, with a mean standard deviation of 0.67. ASK-generated documents thus outperform the average high-quality document in the reference corpus and exhibit consistently higher scores than medium-quality documents across all metrics. Notably, this superior performance is achieved with a lower average word count compared to high-quality reference documents, suggesting that ASK produces more concise yet higher-quality outputs.

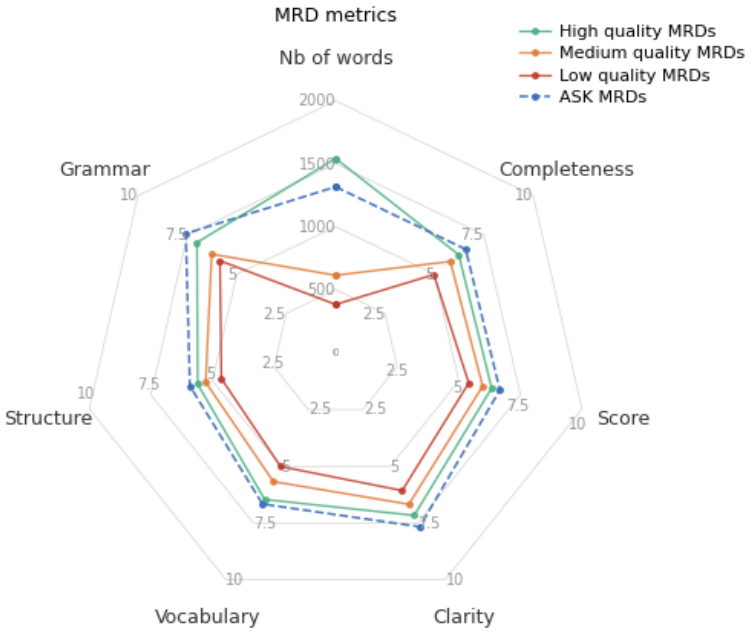

Figure 4: MRDs evaluation metrics results

5. Conclusion and Perspectives

The results of this study suggest that the ASK framework is a promising approach for automating knowledge elicitation. Using the ASK framework, the quality of MRDs improved significantly, surpassing manually written high-quality MRDs on all metrics while having a lower average word count, suggesting better conciseness. The time required to complete them was also substantially reduced compared to manually written MRDs.

Yet several improvements could be made to address the limitations identified during the experimentation. First, providing the Scribe and KC with access to a dynamic memory containing key information gathered throughout the interview—rather than restricting them to the current section and instructions—could improve the relevance and coherence of the KC's guidance. Currently, the KC evaluates requirements sequentially, focusing on the first requirement with a score below four. While this approach preserves the logical flow of questions according to subsection order, it can result in some requirements remaining unmet, causing the

agent to repeatedly ask similar questions. To mitigate this, the system could track previously asked questions and introduce randomized selection of target requirements, reducing redundancy and increasing coverage. Another approach could be to allow the agent to dynamically adjust requirements based on user input. For example, the system could lower the threshold for requirements that the user struggles to fulfill or add new requirements when noteworthy information emerges during the interview. Such adaptability would enhance the agent's responsiveness, better capture relevant knowledge, and further streamline the knowledge elicitation process.

Overall, these refinements could make ASK a more robust and flexible framework, capable of handling the variability of profiles inherent in consulting companies while maintaining high-quality document generation.

Ethics Declaration: The collection of MRDs was an established practice within the organization prior to this study; ASK constitutes a novel means of producing such documents within this existing workflow. All large language models and associated frameworks employed throughout the study were deployed on company-owned servers, ensuring that no data was transmitted to or processed by external systems. Participation in the study was voluntary, and all participants provided informed consent prior to submitting open-ended feedback. In accordance with applicable guidelines governing minimal-risk research, the study was exempt from formal ethical review.

AI Declaration: AI-based tools were used in the preparation of this manuscript to assist with language refinement. Specifically, these tools were used to reformulate and proofread selected sections to improve clarity, grammar, and overall readability. The authors reviewed and validated all outputs to ensure accuracy and integrity of the content.

6. References

Atlassian. 2025. *State of Teams 2025*. Atlassian. https://atlassianblog.wpengine.com/wp-content/uploads/2025/03/the-state-of-teams-2025.pdf (March 9, 2026).

Bubeck, Sébastien, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, et al. 2023. 'Sparks of Artificial General Intelligence: Early Experiments with GPT-4'. https://arxiv.org/abs/2303.12712.

Chen, Banghao, Zhaofeng Zhang, Nicolas Langrené, and Shengxin Zhu. 2025. 'Unleashing the Potential of Prompt Engineering for Large Language Models'. *Patterns* 6(6): 101260. doi:https://doi.org/10.1016/j.patter.2025.101260.

Dwivedi, Yogesh K., Nir Kshetri, Laurie Hughes, Emma Louise Slade, Anand Jeyaraj, Arpan Kumar Kar, Abdullah M. Baabdullah, et al. 2023. 'Opinion Paper: "So What If ChatGPT Wrote It?" Multidisciplinary Perspectives on Opportunities, Challenges and Implications of Generative Conversational AI for Research, Practice and Policy'. *International Journal of Information Management* 71: 102642. doi:https://doi.org/10.1016/j.ijinfomgt.2023.102642.

Freire, Samuel Kernan, Evangelos Niforatos, Chaofan Wang, Sergio Ruiz-Arenas, Mina Foosherian, Stefan Wellsandt, and Alessandro Bozzon. 2023. 'Lessons Learned from Designing and Evaluating CLAICA: A Continuously Learning AI Cognitive Assistant'. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, ACM, 553–68. doi:10.1145/3581641.3584042.

Görer, Binnur, and Fatma Başak Aydemir. 2023. 'Generating Requirements Elicitation Interview Scripts with Large Language Models'. In *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, , 44–51. doi:10.1109/REW57809.2023.00015.

Grand View Research. 2025. *Knowledge Management Software Market Size Report, 2033*. Grand View Research. https://www.grandviewresearch.com/industry-analysis/knowledge-management-software-market-report (March 9, 2026).

Kerrigan, Daniel, Jessica Hullman, and Enrico Bertini. 2021. 'A Survey of Domain Knowledge Elicitation in Applied Machine Learning'. *Multimodal Technologies and Interaction* 5(12): 73. doi:10.3390/mti5120073.

Korn, Alexander, Samuel Gorsch, and Andreas Vogelsang. 2025. 'LLMREI: Automating Requirements Elicitation Interviews with LLMs'. *2025 IEEE 33rd International Requirements Engineering Conference (RE)*: 19–30.

Kuks, Vanessa, Pius Finkel, and Peter Wurster. 2025. 'Potentials, Premises, Perspectives - Using LLMs to Reinterpret Corporate Knowledge Management'. *Industry 4.0 Science* 41(6): 48–56. doi:10.30844/I4SE.25.6.48.

Lee, Jooyeup, Wooyong Jung, and Seungwon Baek. 2024. 'In-House Knowledge Management Using a Large Language Model: Focusing on Technical Specification Documents Review'. *Applied Sciences* 14(5): 2096. doi:10.3390/app14052096.

Malgard, Shiva, Sirous Panahi, Leila Nemati Anaraki, Shadi Asadzandi, and Hossein Ghalavand. 2024. 'The Procedures for Documenting Organizational Knowledge and Experiences: A Scoping Review.' *Medical journal of the Islamic Republic of Iran* 38: 33. doi:10.47176/mjiri.38.33.

Microsoft WorkLab. 2025. '2025: The Year the Frontier Firm Is Born'. https://www.microsoft.com/en-us/worklab/work-trend-index/2025-the-year-the-frontier-firm-is-born (March 9, 2026).

Mihaylov, Dimitar, Massimiliano Vasile, Andrew Herd, Graye Broughton-Stuart, and Yashar Moshfeghi. 2022. 'A Space Conversational Agent for Retrieving Lessons-Learned and Expert Training'. In *Proceedings of the 73rd International Astronautical Congress (IAC 2022)*, Paris, France.

Nonaka, Ikujiro, and Hirotaka Takeuchi. 1995. *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*. New York: Oxford University Press.

Polanyi, Michael. 1966. *The Tacit Dimension*. Garden City, NY: Doubleday.

Rosário, Cláudio Roberto do, and Fernando Gonçalves Amaral. 2021. 'Influence Factors of the Tacit Knowledge Elicitation Process: Systematic Literature Review'. *International Journal of Knowledge Engineering and Management* 9(25): 93–126. doi:10.47916/ijkem-vol9n25-2020-83301.

Singla, Alex, Alexander Sukharevsky, Bryce Hall, Lareina Yee, and Michael Chui. 2025. *The State of AI in 2025: Agents, Innovation, and Transformation*. McKinsey & Company. https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai.