Available online at www.sciencedirect.com

Manufacturing Letters

00 (2026) 000–000

54th SME North American Manufacturing Research Conference (NAMRC 54, 2026)

A Multimodal Manufacturing Safety Chatbot: Knowledge Base Design, Benchmark Development, and Evaluation of Multiple RAG Approaches

Ryan Singh[a], Austin Hamilton[b], Amanda White[a], Michael Wise[c,d], Ibrahim Yousif[c], Arthur Carvalho[a], Zhe Shan[a], Reza Abrisham Baf[c], Mohammad Mayyas[c], Lora A. Cavuoto[e], Fadel M. Megahed[a,*]

[a] *Farmer School of Business, Miami University, 800 E. High Street, Oxford, OH 45056, USA*
[b] *Department of Computer Science and Software Engineering, Miami University, 105 Tallawanda Road, Oxford, OH 45056, USA*
[c] *Department of Engineering Technology, Miami University, 1601 University Blvd., Hamilton, OH 45011, USA*
[d] *Department of Mechanical and Manufacturing Engineering, Miami University, 650 E. High Street, Oxford, OH 45056, USA*
[e] *Department of Industrial and Systems Engineering, University at Buffalo, 407 Bell Hall, Buffalo, NY, 14260*

Abstract

Ensuring worker safety remains a critical challenge in modern manufacturing environments. Industry 5.0 reorients the prevailing manufacturing paradigm toward more human-centric operations. Using a design science research methodology, we identify three essential requirements for next-generation safety training systems: high accuracy, low latency, and low cost. We introduce a multimodal chatbot powered by large language models that meets these design requirements. The chatbot uses retrieval-augmented generation to ground its responses in curated regulatory and technical documentation. To evaluate our solution, we developed a domain-specific benchmark of expert-validated question and answer pairs for three representative machines: a Bridgeport manual mill, a Haas TL-1 CNC lathe, and a Universal Robots UR5e collaborative robot. We tested 24 RAG configurations using a full-factorial design and assessed them with automated evaluations of correctness, latency, and cost. Our top 2 configurations were then evaluated by ten industry experts and academic researchers. Our results show that retrieval strategy and model configuration have a significant impact on performance. The top configuration (selected for chatbot deployment) achieved an accuracy of 86.66%, an average latency of 10.04 seconds, and an average cost of $0.005 per query. Overall, our work provides three contributions: an open-source, domain-grounded safety training chatbot; a validated benchmark for evaluating AI-assisted safety instruction; and a systematic methodology for designing and assessing AI-enabled instructional and immersive safety training systems for Industry 5.0 environments.

Keywords: Cobot safety; Human machine interaction; Industry 5.0; Information retrieval; Smart manufacturing systems

1. Introduction

Manufacturing continues to play a central role in the U.S. economy. In 2024, it generated more than $2.9 trillion in annual GDP, or roughly 10% of total U.S. economic output, according to the Bureau of Economic Analysis [1]. The sector employs over 12.7 million workers [2] and drives around 54% of the research and development conducted by American businesses with 10 or more domestic employees in 2021 [3]. These figures highlight both the economic importance of the sector and the need to support the safety and productivity of its workforce.

During the past 15 years, manufacturing has experienced a significant shift with the rise of smart manufacturing technologies, often referred to as Industry 4.0. These advances include the Industrial Internet of Things (IIoT), cyber-physical systems (CPS), collaborative robots (cobots), machine learning (ML), and other digital tools that provide real-time data for process monitoring and system adaptation [4]. Even with these improvements, Industry 4.0 still faces limitations. Many sys-

[*] Corresponding author. Tel.: +1-513-529-4185.
E-mail address: fmegahed@miamioh.edu (Fadel M. Megahed).

tems operate close to their technological ceiling, where adding more automation produces only small gains. Legacy equipment also makes it difficult to achieve interoperability and reconfigure systems quickly. As production demands change and human robot interaction becomes more common, issues such as ergonomic strain, operator fatigue, and safety concerns continue to appear. These ongoing challenges show that productivity gains alone are not enough to ensure resilience. More recently, Industry 5.0 has emerged to address these concerns by placing human centricity, resilience, and sustainability on equal footing with efficiency [4, 5]. By encouraging closer collaboration between humans and intelligent machines, Industry 5.0 points toward manufacturing systems that are more adaptable, safer, and better able to recover from disruptions while continuing to improve through digital intelligence.

Despite its economic importance, manufacturing continues to face persistent safety challenges. According to the U.S. Bureau of Labor Statistics [6], the sector reported over 320,000 workplace injuries, making it one of the most hazardous occupational domains. Common incidents include cuts, crush injuries, and caught-in or between accidents. These events endanger workers and impose substantial economic costs on employers and the broader economy. The manufacturing sector spends an estimated $7.5 billion each year on direct medical payments and lost wages [7]. Non-fatal workplace injuries also result in an average of 11 lost workdays per employee, equivalent to approximately $1,590 in lost productivity per case [8].

Contributing factors to workplace safety incidents include systems and worker-level factors. Along with the traditional safety concerns, the evolving manufacturing environment has created new challenges for manufacturing operators, as they need to know about and interact with new systems. For example, variations in equipment make it difficult for operators to remember operating procedures across a range of systems. When this is accompanied by the time pressure imposed by high production demands, safety incidents may be more likely to occur.

This is where the application of AI can have a positive influence. For example, prior studies have identify applications of AI in manufacturing to support operators in performing their tasks, including the presentation of maintenance guidelines, process monitoring, predicting machine downtime, and worker training [9, 10]. Modernizing worker training is a critical area for advancing worker performance and mitigating safety risk. Traditional training has often involved classroom or online lecture-based content presented upon job entry and/or at fixed intervals (e.g., annually). This approach is static and fails to reflect the complexity of modern manufacturing environments or worker experiences. In this static setup, when an operator needs information about a piece of equipment, they may not be able to easily access information in a user manual or online. There is thus a pressing need for adaptive, context-aware training systems capable of providing personalized, accurate, and readily accessible safety information in real time.

Following a design science research (DSR) methodology [11], our work identifies three essential requirements that a next-generation manufacturing safety training solution must satisfy: high accuracy, low cost, and low latency. To address these requirements, we propose an open-source Retrieval Augmented Generation (RAG) based chatbot grounded in regulatory documents such as OSHA standards and OEM manuals. Prior work in construction safety shows that chatbot-based training can improve hazard awareness, especially for participants with limited on-site experience [12]. One advantage identified in that work is the ability to tailor training materials to specific worker needs. By combining large language models (LLMs) with curated, domain-specific sources, our RAG system provides precise and contextually relevant safety guidance while maintaining affordability and responsiveness.

A second contribution of our work is the development of a domain-specific benchmark that evaluates the chatbot's safety reasoning capabilities. Our benchmark spans three generations of manufacturing technology. It includes a legacy manually operated system represented by the *Bridgeport Manual Mill*, an automated Industry 3.0 system represented by the *TL-1 CNC* machine, and an Industry 5.0 collaborative system represented by the *Universal Collaborative Robot* (Cobot). We constructed datasets of question and answer pairs related to each machine. All datasets were vetted by industry practitioners and researchers with expertise in smart manufacturing, industrial engineering, occupational safety, and human factors. These benchmark datasets support reproducible evaluation of accuracy and latency and form a foundation for future research in safety-oriented conversational AI systems.

In summary, our work contributes to the growing literature on AI for occupational safety and training by showing how RAG systems can transform static safety instruction into a dynamic/interactive experience. By integrating regulatory grounding, multimodal reasoning, and open benchmarking, our research offers a pathway for using AI to enhance worker safety, compliance, and preparedness in *Industry 5.0* environments.

2. Literature Review

Existing research in manufacturing has explored a range of AI-driven approaches to enhance worker safety, hazard awareness, and on-site decision support. Three interrelated themes emerge from this work. The first is the use of AI, including LLMs and RAG, for training and decision-making. The second is the development of multimodal interfaces, such as vision and language systems, to support richer human-machine interaction. The third is the use of digital twin environments for simulation, monitoring, and operator support. This section reviews representative work in each of these areas.

Conversational AI frameworks have begun to help operators interpret real-time industrial data. For example, Jeon et al. [13] observe that limited digital literacy among shop-floor workers often makes conventional monitoring interfaces difficult to use. Their system, *ChatCNC*, addresses this issue by allowing operators to ask plain-language questions about machine conditions, such as whether a machine is overheating, instead of navigating complex software menus and interfaces.

Research in this space reflects a broader shift toward AI-mediated instruction and monitoring that aims to reduce human error and improve safety outcomes. Natural language techniques such as LLMs are particularly attractive because they blend general reasoning capability with a conversational interface that aligns well with Industry 5.0's emphasis on human-centric smart manufacturing. For example, Wang et al. [14] developed a collaborative robot with vision–language abilities that enable it to visually map its surroundings and follow spoken or written instructions through an LLM-based parser. Similarly, Lou et al. [15] built an LLM-powered cognitive agent that plans and makes decisions for assembly and disassembly tasks. Their agent demonstrates context awareness and outperforms traditional automation by leveraging the LLM's extensive knowledge and reasoning abilities. These examples show how LLMs can function as conversational intermediaries that translate operator intentions into complex technical actions.

A major challenge for LLMs in manufacturing is the gap between broad model training and the specific, real-time knowledge needed on the factory floor. Retrieval-Augmented Generation (RAG) has emerged as a practical solution to this issue by pairing an LLM with a retrieval component that supplies relevant domain information such as documents, sensor readings, or knowledge base entries. Fan et al. [16] illustrate this approach with `MaViLa`, a vision–language model capable of interpreting live visuals and grounding its guidance with retrieved manufacturing knowledge. While `MaViLa` focuses on broad manufacturing tasks such as process understanding and skill acquisition, our work targets manufacturing safety and uses RAG to ground responses in a curated safety knowledge base.

Another important development in manufacturing involves the growing adoption of multimodal interfaces that integrate vision, language, and other data modalities. Many shop-floor scenarios are inherently visual, such as identifying hazards or verifying proper use of protective equipment, which means an AI assistant that can both see and converse offers clear advantages over text-only systems. The industrial metaverse described by Li et al. [17] envisions manufacturing environments where workers and AI interact through virtual and augmented reality tools that combine digital avatars, spatial audio, mixed-reality visuals, and text. These trends demonstrate that an effective manufacturing chatbot must be multimodal so that it can process images or sensor data and generate language that supports safety monitoring and operator assistance. Multimodality therefore serves as a core design principle in our work.

Digital twins and virtual environments have also emerged as key platforms for integrating LLMs, RAG, and multimodal interfaces in realistic, safety-oriented settings. A digital twin is a dynamic virtual representation of a physical system that maintains continuous two-way communication with its real-world counterpart, enabling real-time monitoring, predictive simulation, control, and what-if analysis. In manufacturing, these technologies support proactive safety management and training without interrupting production. Gautam et al. [18] demonstrate this potential by linking an LLM-based agent system with a factory digital twin. Their system uses specialized LLM agents for machine expertise, data visualization, and fault diagnosis to process streaming IIoT data and reflect the factory state in a virtual model. The digital twin then offers an interactive 3D interface where operators can query an avatar about system status or troubleshooting steps, allowing them to gain insights without approaching hazardous equipment.

Taken together, these developments show the main components of a modern manufacturing safety assistant. They demonstrate that LLMs support natural communication, that RAG provides timely and relevant domain knowledge, and that multimodal understanding is essential for interpreting factory conditions. Based on these insights, we created a retrieval-augmented multimodal chatbot for manufacturing safety training and hazard prevention. Our system uses an extensive safety knowledge base, perceives the manufacturing environment, and engages in dialog to deliver practical safety guidance and real-time risk analysis.

3. Research Methodology

We address the challenge of modernizing hazard-recognition training in manufacturing by adopting a design science research methodology (DSR) [11]. This methodological approach is well-suited for the development and evaluation of information technology solutions (artifacts) that aim to solve complex, real-world problems. Following DSR ensures that our work systematically integrates scientific rigor with practical relevance, producing outcomes that are both theoretically grounded and operationally viable within industrial settings.

In particular, our research process follows the six-stage model outlined by Peffers et al. [11], which we elaborate on throughout the paper: (1) problem identification and motivation, (2) definition of solution objectives, (3) artifact design and development, (4) demonstration, (5) evaluation, and (6) communication. We have already identified and motivated the problem in the preceding section, and the last stage of the DSR methodology, i.e., communication, is represented by this paper. That said, we now turn to the second stage, namely defining the solution objectives, which are derived "*from the problem definition and knowledge of what is possible and feasible*" [11]. To operationalize this stage, we established a set of *design requirements* that any proposed artifact must satisfy to address the underlying hazard-recognition and safety-training challenges.

Our first design requirement concerns system *accuracy*, which represents the non-negotiable foundation of any effective safety-training system. In industrial environments, inaccurate or misleading instructions can lead to improper procedures, equipment misuse, and, in the worst cases, fatal accidents. A training artifact that compromises factual precision risks undermining the very purpose of hazard education. Therefore, every element of the system, whether textual guidance, visual cues, or procedural demonstrations, must convey information that is technically correct, operationally safe, and fully aligned with regulatory and manufacturer standards. The following requirement definition captures the above discussion.

> **Design Requirement #1 (Accuracy)**: Training artifacts must deliver information that is technically correct and operationally safe.

Our second design requirement concerns system *latency*, reflecting the need for timely and responsive feedback within dynamic manufacturing environments. In industrial contexts, hazards can emerge and evolve in seconds, and delayed or sluggish responses can render safety guidance ineffective or even dangerous. A training system that fails to deliver instructions, alerts, or assessments in real time risks losing its relevance to the unfolding situation on the shop floor. Therefore, any proposed solution must minimize latency in both data processing and user feedback, ensuring that workers receive immediate, context-appropriate information as they interact with machinery or simulated environments. Low latency is essential not only for maintaining user engagement and situational awareness but also for reinforcing procedural learning under realistic temporal conditions. The following definition summarizes our second design requirement.

> **Design Requirement #2 (Latency)**: Artifacts must provide timely and responsive feedback, ensuring that safety instructions are rapidly provided under changing conditions in manufacturing environments.

Our third design requirement concerns the *cost* of deployment and operation, emphasizing the importance of economic accessibility for broad adoption across the manufacturing sector. Safety innovation cannot achieve its intended societal impact if the associated technologies are prohibitively expensive to implement, maintain, or scale. Many small and medium-sized manufacturers, which account for 98 percent of U.S. manufacturing firms [19], face constrained capital and “do not have the resources” to invest in advanced technologies and associated training [20]. Therefore, training solutions must minimize recurring expenses such as software licensing, hardware requirements, and data-processing costs, without compromising instructional quality or safety integrity.

> **Design Requirement #3 (Cost)**: Artifacts must minimize deployment and operational costs to ensure affordability and, consequently, adoption across manufacturers of varying sizes.

In the following sections, we design, demonstrate, and evaluate an artifact that satisfies the above design requirements. We do so by proposing the six-phase framework in Figure 1 for building manufacturing-safety chatbots grounded in expert-curated operational and safety documents. These phases align perfectly with our DSR methodology. For example, Phases 1 and 2 are part of the third DSR stage called “artifact design and development.” In particular, in Phase 1, we curate and preprocess safety materials from OSHA, the National Institute for Occupational Safety and Health (NIOSH), and Original Equipment Manufacturer (OEM) manuals to build a reliable, structured corpus. In Phase 2, we construct multiple knowledge bases that integrate keyword, semantic, and graph indices to support flexible information retrieval for question answering (Q&A). In Phase 3, we transition to the fourth design science stage, namely artifact demonstration. In particular, we demonstrate how a grounded, multimodal chatbot can be developed based on the knowledge base and indexing from the previous phase. Phases 4 to 6 are linked to the fifth stage of our DSR methodology, namely artifact evaluation. Phase 4 develops a benchmark Q&A dataset, with expert-validated questions and reference answers. We evaluate multiple retrieval-augmented generation (RAG) configurations using automated metrics in Phase 5 to identify a reasonable subset of configurations for human evaluation. In Phase 6, we employ blind human evaluation to select the “best” configuration. The evaluations in Phases 4 to 6 enable us to implement a single configuration and deploy a multimodal chatbot that integrates text, visual, and speech interactions, ensuring that its responses remain grounded in our curated source documents.

4. Artifact Design and Development

In the third stage of our DSR methodology, we designed and developed a concrete artifact. While many digital solutions could theoretically address the challenge of delivering accessible, accurate, and adaptive safety training in manufacturing, few can simultaneously meet the requirements of accuracy, low latency, and affordability identified in Section 3. For example, generic e-learning platforms or static digital manuals often fall short because they lack contextual awareness, interactivity, and real-time responsiveness. In contrast, modern AI technologies, such as a multimodal chatbot capable of processing text, images, and speech, offer a uniquely flexible and human-centered medium for safety communication. In particular, by embedding RAG techniques within a multimodal interface, such a system can ground its responses in authoritative safety documents while allowing workers to query information naturally through the mode most convenient to their environment. This convergence of adaptive retrieval, multimodal reasoning, and conversational accessibility positions a RAG-based chatbot as a suitable artifact to operationalize the design requirements and realize the goals of Industry 5.0–aligned, human-centric safety training. We next describe the first two phases of our development framework (Figure 3), which focus on data curation and the construction of RAG methods.

4.1. Phase 1: Source Curation and Corpus Development

Phase 1 establishes the foundation of the chatbot development framework by focusing on the systematic curation and preparation of safety-critical source materials. This stage ensures that all subsequent modeling and retrieval processes are grounded in authoritative, high-quality data. It involves collecting and organizing diverse safety documents into a unified corpus suitable for RAG. This phase includes four steps: (1) collecting the source documents, (2) extracting and cleaning text, (3) segmenting the content into meaningful units, and (4) standardizing how each segment is saved.

4.1.1. Collect Safety Documents

Grounding LLM responses in authoritative documents reduces hallucinations and improves factual correctness when compared to generation from the LLM alone [21]. We curated

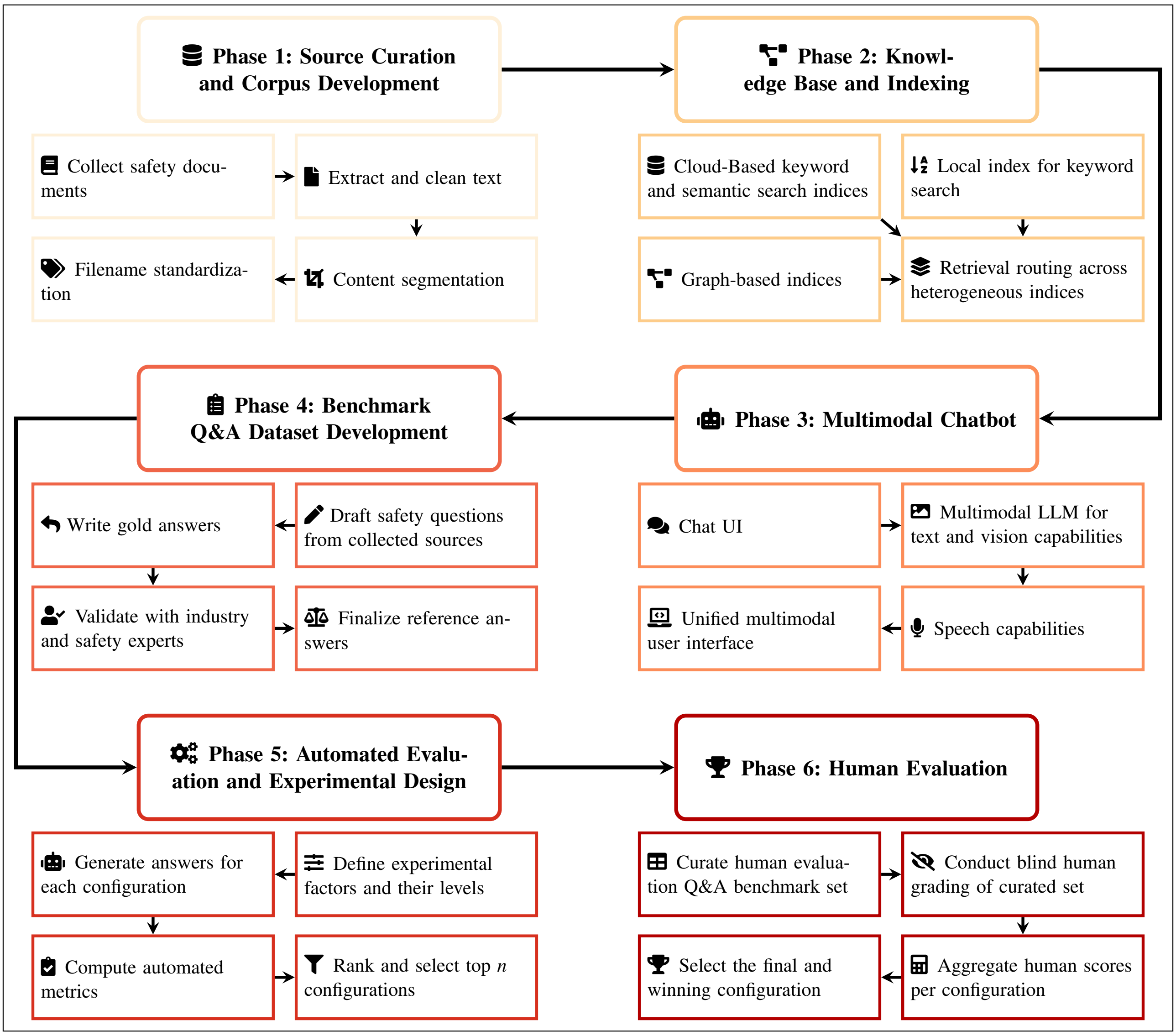

Phase 1: Source Curation and Corpus Development
Collect safety documents
Extract and clean text
Filename standardization
Content segmentation
Phase 2: Knowledge Base and Indexing
Cloud-Based keyword and semantic search indices
Local index for keyword search
Graph-based indices
Retrieval routing across heterogeneous indices
Phase 4: Benchmark Q&A Dataset Development
Write gold answers
Draft safety questions from collected sources
Validate with industry and safety experts
Finalize reference answers
Phase 3: Multimodal Chatbot
Chat UI
Multimodal LLM for text and vision capabilities
Unified multimodal user interface
Speech capabilities
Phase 5: Automated Evaluation and Experimental Design
Generate answers for each configuration
Define experimental factors and their levels
Compute automated metrics
Rank and select top n configurations
Phase 6: Human Evaluation
Curate human evaluation Q&A benchmark set
Conduct blind human grading of curated set
Select the final and winning configuration
Aggregate human scores per configuration

Figure 1. An overview of our proposed six-phase framework for developing, demonstrating, and evaluating a multimodal manufacturing-safety chatbot.

four categories of documents to ensure that the chatbot's responses are grounded in authoritative safety expectations: (1) OSHA Laws and Regulations, i.e., CFR 1910 Subparts O and N, (2) OSHA technical guidance documents, (3) NIOSH safety alerts, and (4) OEM manuals for our target machines. Figure 2 summarizes our data sources.

4.1.2. Extract and Clean Text

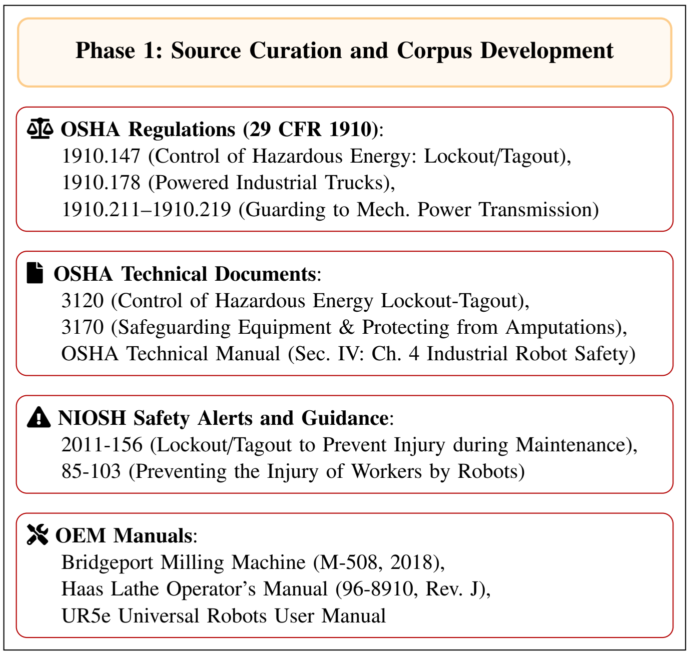
For each document, we first determined whether it was machine-readable (i.e., searchable or selectable in a standard PDF reader). For machine-readable PDFs, we used the `PyMuPDF` library in Python [22], which returns page-level text while preserving paragraph structure. If the document stored text as embedded images, we applied the Mistral OCR API [23] to produce a text layer with page boundaries and paragraph breaks. After OCR, these documents were processed identically to machine-readable files, ensuring a consistent representation across the corpus.

4.1.3. Content Segmentation

For longer documents, such as OSHA standards and OEM manuals, we used the table of contents to identify the start of chapters and subchapters, and split the document accordingly. This segmentation keeps each text unit aligned with meaningful instructional topics. Shorter narrative documents, such as NIOSH safety alerts, often lack a usable table of contents. For these, we applied a length-based rule: documents exceeding 3,000 words were divided into smaller sections. This threshold balances retrieval performance by preventing segments from becoming excessively large or too fragmented.

4.1.4. Filename Standardization

Once segmented, each section was saved as its own PDF, and any repeated header, footer, or boilerplate content was removed when present. We standardized filenames to include the source type (e.g., OSHA, NIOSH, OEM), the document identifier, the section title, and the original page range. This naming conven-



Phase 1: Source Curation and Corpus Development

⚖ **OSHA Regulations (29 CFR 1910)**:
1910.147 (Control of Hazardous Energy: Lockout/Tagout),
1910.178 (Powered Industrial Trucks),
1910.211–1910.219 (Guarding to Mech. Power Transmission)

🗎 **OSHA Technical Documents**:
3120 (Control of Hazardous Energy Lockout-Tagout),
3170 (Safeguarding Equipment & Protecting from Amputations),
OSHA Technical Manual (Sec. IV: Ch. 4 Industrial Robot Safety)

⚠ **NIOSH Safety Alerts and Guidance**:
2011-156 (Lockout/Tagout to Prevent Injury during Maintenance),
85-103 (Preventing the Injury of Workers by Robots)

🛠 **OEM Manuals**:
Bridgeport Milling Machine (M-508, 2018),
Haas Lathe Operator's Manual (96-8910, Rev. J),
UR5e Universal Robots User Manual

Figure 2. Regulatory, safety, and OEM documents forming our safety corpus.

tion ensures that each file is traceable back to its authoritative source and simplifies manual review, debugging, and alignment in downstream retrieval steps.

4.2. Phase 2: Knowledge Base and Indexing

The goal of this phase is to construct the knowledge bases from the segmented documents in Phase 1 and develop the retrieval mechanisms that the chatbot relies on when answering questions. Prior research has shown that different retrieval strategies can surface different types of relevant content [21]. Because manufacturing safety questions may use precise terminology or broader hazard descriptions, we examined multiple retrieval methods rather than relying on a single strategy. In later phases, we evaluate which approach performs best when answering the benchmark questions developed in Phase 4.

4.2.1. Cloud-Based Keyword and Semantic Search Indices

We uploaded the segmented PDF documents from Phase 1 into an OpenAI-managed vector store, i.e., a database that stores documents as numerical representations (embeddings), enabling comparison and retrieval based on similarity in a high-dimensional vector space. OpenAI was selected as the vector store and retrieval provider for two main reasons. First, its embedding models and retrieval API are tightly integrated, reducing engineering overhead and limiting potential sources of error in data preprocessing and indexing. Second, OpenAI provides a fully managed, scalable service with strong latency and reliability guarantees, simplifying deployment and supporting real-time interactive querying.

During ingestion, we specified a chunk size of 4,000 tokens (approximately 3,000 words; the default is 800 tokens) to ensure that the API did not further subdivide our segmented documents. The `OpenAI Retrieval API` supports two retrieval modes over the same stored data [24]. In *keyword search* mode, the system prioritizes exact or near-exact lexical matches between the query terms and the stored content. In *semantic search* mode, the system rewrites or embeds the query and retrieves passages whose vector representations are close in meaning, even when they do not share surface-level wording. Keyword search is most effective when users know and reuse technical terms verbatim, whereas semantic search better supports users who describe their information needs in everyday language or with partial recall of the original phrasing.

4.2.2. Local Index for Keyword Search

In addition to the cloud-based indices, we constructed a local keyword-based index using the BM25 retriever in the Python `LangChain` library [25]. BM25 employs probabilistic term weighting to rank documents based on estimated relevance and has remained a competitive baseline in information retrieval [26]. We selected BM25 for three primary reasons. First, as a purely lexical method, BM25 is well aligned with queries that contain distinctive regulatory language, component identifiers, or procedural step names, all of which tend to appear verbatim in technical manuals. Second, maintaining the BM25 index locally eliminates per-query API costs and reduces dependency on external services, which is advantageous for repeated experimentation and for potential deployment in cost-sensitive or bandwidth-constrained environments. Third, local execution provides consistently low latency and full control over the indexing and retrieval pipeline, facilitating reproducibility and fine-grained inspection of retrieval behavior.

4.2.3. Graph-Based Indices

Graph-based retrieval methods allow the system to use both textual similarity and structural relationships among safety concepts, procedures, and component interactions. This is especially valuable in technical domains where relevant information spans multiple steps or subcomponents, and where safe outcomes depend on understanding dependencies rather than isolated passages. Following the `LangChain` `GraphRetriever` design [27], we implemented three graph RAG variants that differ in how they explore or constrain graph neighborhoods. *AstraDB* serves as the online vector storage backend for all three configurations [28] because it provides scalable embedding storage with native support for vector similarity search and integrates smoothly with graph-based retrieval workflows.

The three variants provide complementary strategies. The `Graph Eager` retriever expands outward through connected nodes to gather broader contextual information, which helps when queries are imprecise. The `Graph MMR` variant applies maximal marginal relevance to balance relevance and diversity, reducing redundancy across similar segments. Finally, the vanilla similarity search baseline over *AstraDB* performs traditional k-nearest-neighbor retrieval without graph traversal and serves as a reference for assessing the added value of graph-based methods.

4.2.4. Retrieval Routing Across Heterogeneous Indices

We implemented a unified retrieval router to coordinate access to the six retrieval mechanisms described above: (1) OpenAI keyword search, (2) OpenAI semantic search, (3) the local BM25 retriever, (4) Graph Eager, (5) Graph MMR, and (6) vanilla similarity search over *AstraDB*. The router decouples retrieval strategy selection from the rest of the conversational pipeline by standardizing how results are formatted and passed to the underlying LLM. This ensures that downstream evaluation compares retrieval strategies fairly and that performance differences can be attributed to retrieval behavior rather than formatting variations. Operationally, the routing layer allows retrieval strategies to be swapped in or out without modifying the chatbot architecture, enabling controlled experimentation in Phases 4 to 6 and simplifying demonstration in Phase 3.

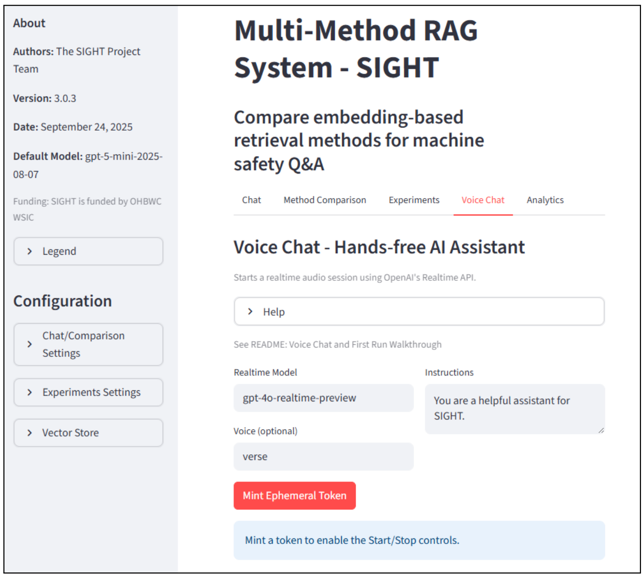
5. Artifact Demonstration

The fourth stage of our DSR methodology is to "*demonstrate the use of the artifact to solve one or more instances of the problem*" [11]. For our case, this involved deploying a publicly accessible multimodal chatbot interface that exposes the safety-assistance capabilities of the underlying RAG system to end users. The system focuses on three types of equipment: the Bridgeport Manual Mill, TL-1 CNC Machine, and the Universal Robots Collaborative Robot (Cobot). The interface supports three input modes: (1) text-based dialogue, (2) image-based queries, and (3) speech-to-speech interaction.

Operationally, the fourth stage corresponds to Phase 3 of our development framework. We began by developing a text chat interface that allows workers to pose safety-related questions in everyday language. The system then retrieves and synthesizes answers grounded in the authoritative documents curated earlier in the pipeline (see Section 4.1). We next added an image input mode, enabling users to upload photographs of equipment, controls, or workspace conditions. The model interprets the visual context and issues a retrieval query tailored to the situation, returning relevant safety passages and procedures. Finally, we added a speech modality to support hands-free interaction, allowing users to speak their questions and receive spoken responses.

The deployed system was implemented as a Streamlit application [29] that integrates OpenAI's Responses and Realtime APIs to provide text, image, and speech capabilities through a unified interface. Retrieval operates over the same corpus, vector stores, and indices established in earlier phases, preserving traceability between responses and their documentary sources. The application also exposes the retrieval router described in Section 4.2.4, allowing users and researchers to select among different RAG configurations and to experiment with them in situ. This capability supports both practitioner use and systematic comparison in later evaluation phases.

The system logs session-level analytics (e.g., query types, retrieval configurations, and response characteristics) and provides transparent citations linking each answer to specific source passages, supporting explainability, auditability, and user trust. A screenshot of the deployed application is shown in Figure 3, and the full code implementation is available in the project repository (see "Data and Code Availability" section).



About

Authors: The SIGHT Project Team

Version: 3.0.3

Date: September 24, 2025

Default Model: gpt-5-mini-2025-08-07

Funding: SIGHT is funded by OHBWC WSIC

Configuration

Multi-Method RAG System - SIGHT

Compare embedding-based retrieval methods for machine safety Q&A

Voice Chat - Hands-free AI Assistant

Starts a realtime audio session using OpenAI's Realtime API.

See README: Voice Chat and First Run Walkthrough

Realtime Model

gpt-4o-realtime-preview

Instructions

You are a helpful assistant for SIGHT.

Voice (optional)

verse

Mint a token to enable the Start/Stop controls.

Figure 3. A screenshot of the voice module of our deployed chatbot. The chatbot is available at https://sight.fsb.miamioh.edu/.

6. Artifact Evaluation

The fifth stage of our DSR methodology comprises "*comparing the objectives of a solution to actual observed results from use of the artifact in the demonstration*" [11]. In our context, this stage corresponds to a systematic assessment of how well the proposed RAG configurations meet the design requirements identified earlier, namely high accuracy, low latency, and cost. To enable this assessment, we adopt a three-stage evaluation pipeline within our development framework. In Phase 4, we curate a domain-specific benchmark of safety-related queries and ground-truth reference answers derived from the authoritative corpus. In Phase 5, we use the above benchmark to conduct controlled experiments over multiple RAG configurations. Finally, in Phase 6, we conduct expert-based subjective evaluations to evaluate the most promising configurations, ultimately narrowing down to a single configuration that best balances performance, cost, and latency for deployment.

6.1. Phase 4: Benchmark Q&A Dataset Development

To evaluate the performance of the RAG systems, we developed a comprehensive benchmark consisting of questions and gold-standard answers derived from the operational and safety manuals of the three target machines. This benchmark was designed to assess the chatbot's ability to retrieve, reason about, and accurately convey safety-critical information across diverse levels of difficulty.

The process began with our research team conducting a detailed review of each machine's official manuals. From these materials, we systematically extracted and formulated questions that represent realistic scenarios faced by operators and trainees in manufacturing settings. Each question was paired with a corresponding answer grounded explicitly in the manuals or regulatory guidelines, ensuring factual correctness and traceability. For instance, an example of an easy question might be: `What is a pinch hazard in robot operation?` with the corresponding answer: `A pinch hazard occurs when body parts can be caught between moving robot parts or surfaces.` Easy questions can typically be answered from a single section of a manual. To better simulate realistic industrial information retrieval tasks, we also created more challenging questions that required integrating knowledge from multiple sections or documents. For example, a complex question might ask: `What are the external connection ports on the robot?` The answer should be `Teach Pendant Port, SD Card slot, Ethernet, USB 2.0 and USB 3.0, Mini Displayport, 10A Mini Blade Fuse`, which requires synthesizing details scattered across different technical sections. This ensured the benchmark tested both direct retrieval and reasoning capabilities of the system.

After the initial Q&A pairs were derived, we implemented a validation phase to ensure content accuracy and industrial relevance. Each question and answer was reviewed either by experienced operators and/or safety professionals from partner companies as well as by researchers with expertise in manufacturing, engineering technology, occupational safety and human factors engineering. Their feedback was used to refine the clarity, technical precision, and contextual relevance of the benchmark items. This iterative validation process guaranteed that the benchmark accurately represented the complexity of real-world safety knowledge and could be reliably used for evaluating AI system performance across different machines and hazard contexts. We refer to the final, validated answers in our benchmarks as *gold-standard answers*.

In total, we created 60 questions for the `UR5 Cobot`, 51 for the `Haas TL-1 lathe`, and 42 for the `Bridgeport Manual Mill`. Due to space constraints, we henceforth focus on the `UR5 Cobot` in our evaluations since it exhibits greater operational complexity and a higher degree of human–machine interaction than the other two machines. We argue that the `UR5` benchmark sufficiently tests the system's ability to retrieve and synthesize information across multiple text segments ("chunks") produced during corpus pre-processing. As such, including additional results for other machines would provide little to no insight beyond what is demonstrated through the `Cobot` evaluation.

6.2. Phase 5: Automated Evaluation and Experimental Design

We used a full-factorial experimental design to study how retrieval method and model specification choices influenced the quality of answers generated by the AI. The purpose was to identify which configuration patterns consistently yielded correct responses obtained fast and cheaply before advancing a small number of candidate configurations for human evaluation in Phase 6. In particular, we vary three factors, summarized in Table 1, resulting in 24 retrieval-generation *pipelines*. Henceforth, we use the term pipeline to refer to a specific combination of the three factor levels used to generate an answer from the LLM/chatbot. Each pipeline is evaluated across all 60 benchmark questions, yielding $24 \times 60 = 1,440$ generated responses per evaluation metric.

Table 1. Factors and levels in the full-factorial experimental design.

Factor	Purpose	Levels
Retrieval Approach	Determines how relevant safety text is selected	{`OpenAI Keyword`, `OpenAI Semantic`, `BM25`, `Graph Eager`, `Graph MMR`, `Vanilla`}
LLM Used	Controls model capacity and reasoning ability	{`gpt-5-mini-2025-08-07`, `gpt-5-nano-2025-08-07`}
Max Output Tokens	Constrains verbosity and potential reasoning detail for generated responses	{5000}, a fixed factor to simplify our experiment
Reasoning Effort	Sets structured reasoning depth for LLM	{*low*}, a fixed factor to simplify our experiment
Top−k Retrieval Depth	Number of chunks returned	{3, 7}
Total Pipelines = $6 \times 2 \times 1 \times 1 \times 2 =$ **24**		

For each pipeline, we generate an answer to every benchmark question using that pipeline's specific settings. All outputs are stored together with their pipeline, question, and replicate identifiers to support consistent evaluation. We used a single system prompt (set of instructions) in our experiments. We set the underlying LLM's temperature parameter to 0 to reduce the randomness from the sampling process, making responses more deterministic for a given pipeline.

Given the generated answers from each pipeline alongside the gold-standard answers, we computed a set of evaluation metrics corresponding to the three design requirements in Section 3. The complete set of metrics and their definitions are summarized in Table 2. First, accuracy was assessed using an LLM-as-judge that compares each generated answer to the gold-standard answer and returns a binary decision. The LLM-based judgment used standardized evaluation prompt templates provided in the LangChain "RAG Evaluation" tutorial [30] and ChatGPT 5 as the model evaluator. Second, latency was measured as the end-to-end response time for answer generation. Third, cost values were built into the experiment design by restricting model selection to smaller/distilled versions of ChatGPT 5, namely `gpt-5-mini-2025-08-07` and `gpt-5-nano-2025-08-07`. These LLMs are considered to be high-performing and relatively low-cost. As of October 2025, the `gpt-5-mini-2025-08-07` was priced at \$0.25/1M input tokens and \$2/1M output tokens, and `gpt-5-nano-2025-08-07` was priced at \$0.050/1M input tokens and \$0.400/1M output tokens [31].

6.3. Phase 6: Human Evaluation

While automated evaluations provide valuable initial insights, they can only go so far in identifying the best-performing retrieval–generation pipeline. For example, metrics such as LLM-as-judge may be imperfect for distinguishing accuracy

Table 2. Summary of evaluation metrics used in our framework/experiment.

Metric	Definition / Prompt Basis
Accuracy	Binary decision from LLM-as-judge comparing the generated answer with its corresponding gold-standard answer
Latency	Model response time
Cost	Token usage cost for `gpt-5-mini` and `gpt-5-nano`

across competing solutions. To complement these quantitative results and select a single final pipeline for potential deployment, we conducted a human expert evaluation.

In the first stage, we evaluated all 24 pipelines on the full `UR5 Cobot` benchmark. Performance was assessed primarily on accuracy. Latency was considered as a secondary criterion, and cost was treated as an acceptability threshold rather than a ranking factor. When pipelines had no practical differences in accuracy or latency, we relied on their accuracy ranking to guide the selection of candidates. Based on this full-benchmark evaluation, we narrowed the original set of 24 pipelines down to 2.

After this narrowing step, we drew a random sample of ten questions from the `UR5 Cobot` benchmark for human evaluation. For each sampled question, we retrieved the corresponding answers generated by the two selected pipelines. These ten questions and answers were then presented to expert evaluators through a purpose-built web interface that enabled side-by-side comparison of model outputs. Figure 4 shows this evaluation interface. This setup allowed us to distribute a structured survey to domain specialists in manufacturing, occupational safety, and human factors.

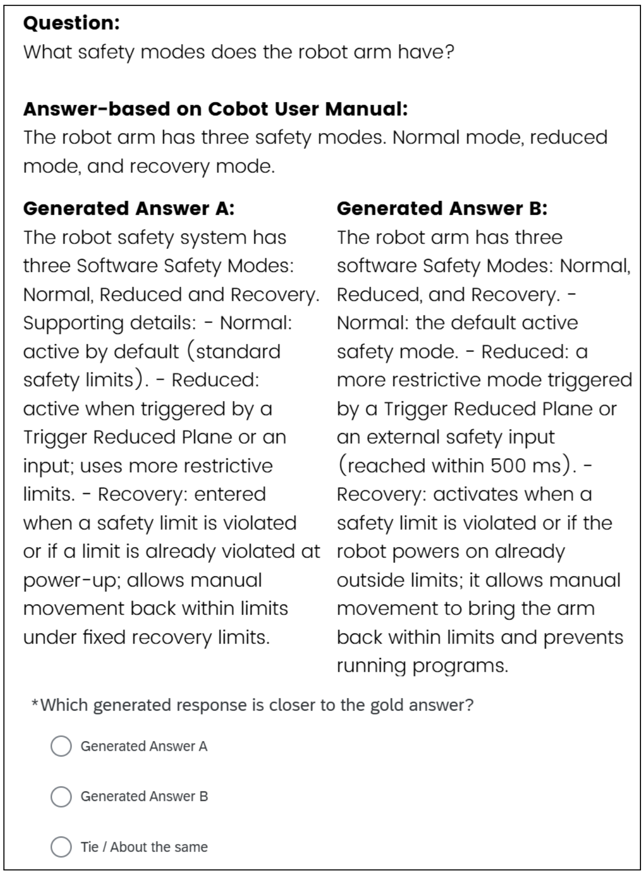
6.4. Results

We next report our experimental results by comparing different RAG configurations to understand their impact on the metrics of interest. Specifically, we highlight key trends and tradeoffs observed in our experiments, leading to the configuration that offers the most practical balance for deployment in manufacturing training settings. Figure 5 summarizes our results.

6.4.1. Accuracy

First, we examined how each of the pipelines affected the proportion of answers judged correct relative to the gold-standard answers (as depicted in Figure 5A). Our reported analyses in this subsection focus on main effects. That is, the average impact of each factor when collapsing over the others, rather than on specific interactions between model, retrieval depth, and retrieval method.

Across all conditions, the larger `gpt-5-mini-2025-08-07` model achieved a higher overall correctness rate (0.73) than the smaller `gpt-5-nano-2025-08-07` model (0.62). A paired McNemar test confirmed that this difference was statistically significant, $\chi^2(1, N = 720) = 19$, $p < .001$. In other words, the presumed additional capacity of the `mini` variant translated into more accurate safety-related answers in our benchmark when averaging across retrieval depths and approaches.



Question:
What safety modes does the robot arm have?

Answer-based on Cobot User Manual:
The robot arm has three safety modes. Normal mode, reduced mode, and recovery mode.

Generated Answer A:
The robot safety system has three Software Safety Modes: Normal, Reduced and Recovery. Supporting details: - Normal: active by default (standard safety limits). - Reduced: active when triggered by a Trigger Reduced Plane or an input; uses more restrictive limits. - Recovery: entered when a safety limit is violated or if a limit is already violated at power-up; allows manual movement back within limits under fixed recovery limits.

Generated Answer B:
The robot arm has three software Safety Modes: Normal, Reduced, and Recovery. - Normal: the default active safety mode. - Reduced: a more restrictive mode triggered by a Trigger Reduced Plane or an external safety input (reached within 500 ms). - Recovery: activates when a safety limit is violated or if the robot powers on already outside limits; it allows manual movement to bring the arm back within limits and prevents running programs.

*Which generated response is closer to the gold answer?

○ Generated Answer A

○ Generated Answer B

○ Tie / About the same

Figure 4. A screenshot of the interface used for human evaluation.

Next, we assessed the effect of the number of retrieved chunks provided to the LLM (`top k` = 3 vs. 7). Accuracy was similar across these conditions, with correctness proportions of 0.67 and 0.68 for `top k` = 3 and `top k` = 7, respectively. McNemar's test found no statistically significant difference between these retrieval-depth settings, $\chi^2(1, N = 720) = 38$, $p = 0.445$. From a practical perspective, correct proportions of 0.67 and 0.68 are also practically indistinguishable, which implies that the added computational cost of retrieving more chunks may not be justified in our case.

Finally, we investigated the main effect of retrieval approach on answer accuracy. Cochran's Q test indicated a significant effect of retrieval method, $Q(5) = 103.24$, $p < 0.001$. Averaged across models and retrieval depths, the highest proportions of correct answers were observed for the `OpenAI Keyword` and `OpenAI Semantic` approaches (both 0.77), followed by `BM25` (0.71), `Vanilla` (0.68), `Graph MMR` (0.62), and `Graph Eager` (0.50). Post-hoc pairwise McNemar tests with Holm adjustment, focusing on the three best-performing methods, revealed no significant differences among `OpenAI Keyword`, `OpenAI Semantic`, and `BM25` (all adjusted $p = 0.2$), indicating statistically equivalent performance at the top end. Both OpenAI-based methods, however, significantly outperformed all remaining baselines, namely `Graph Eager`, `Graph MMR`,

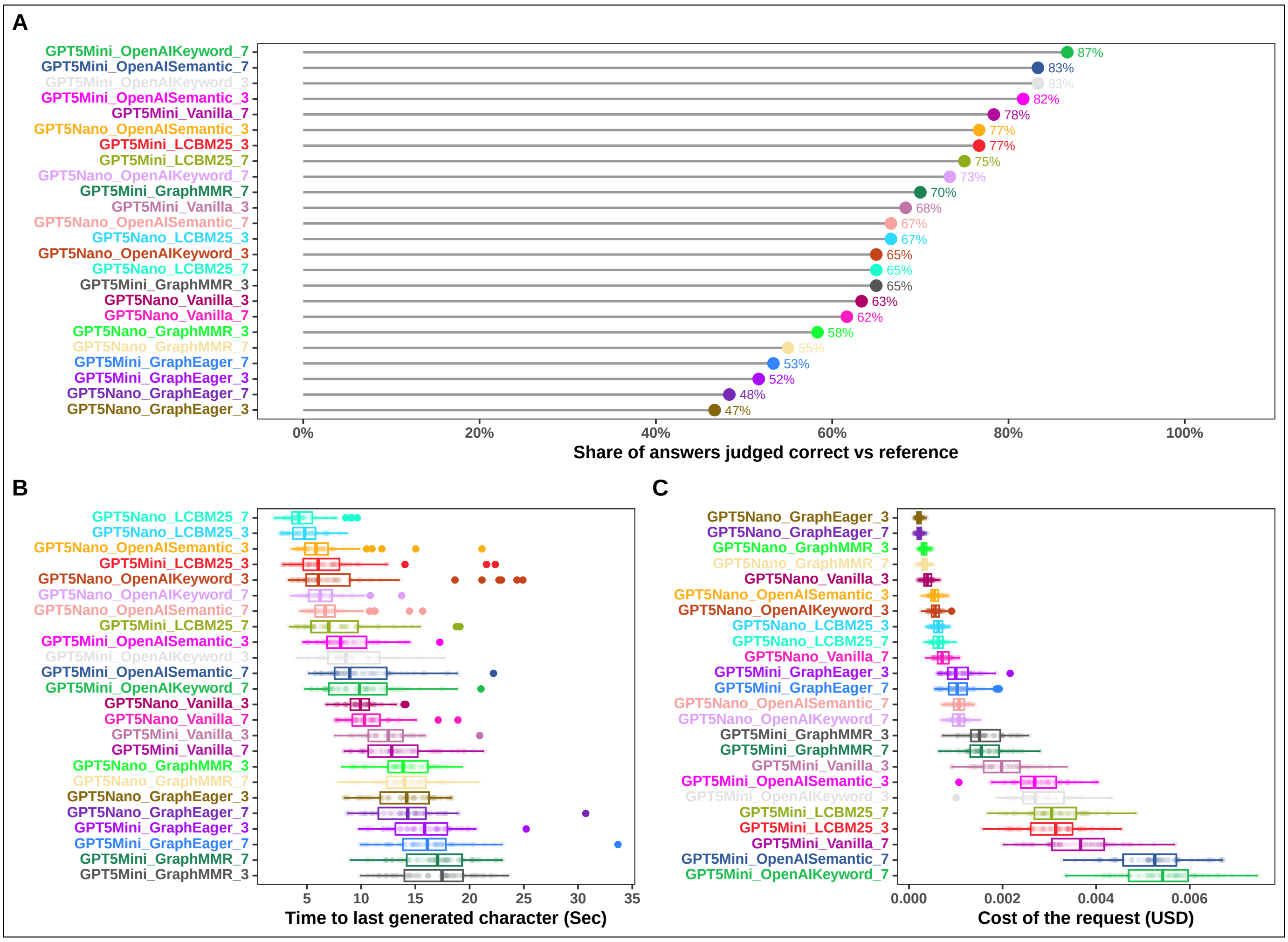


Figure 5. Comparison of the performance of our 24 RAG pipelines across correctness (A), generation time, i.e., latency (B), and cost (C). The y-axis in each subplot utilizes the `LLMModel_RagApproach_TopKChunksUsedAsInput` naming convention. For each subfigure, we sort the pipelines by performance, placing the best at the top (using median performance for the box plots).

and `Vanilla` (all adjusted $p < 0.05$), highlighting the advantage of these learned retrieval strategies for maximizing response correctness.

It is important to note that these results characterize average (main) effects of each factor. That is, we ask questions such as "Is `mini` more accurate than `nano` on average?" or "Which retrieval approach performs best on average?" rather than examining the full pattern of interactions (for example, whether certain retrieval methods benefit `mini` more than `nano`, or whether `top k` = 7 helps some methods but not others). Analyses at the level of specific model–method–depth combinations, such as those visualized in Figure 5A, would be required to fully characterize such interaction effects.

6.4.2. Latency

We next examined how each of the model choice, retrieval depth, and retrieval approaches influenced end-to-end response time (Figure 5B). As expected, the smaller `gpt-5-nano-2025-08-07` model generated responses more quickly than the larger `gpt-5-mini-2025-08-07` model. Mean latency for the `mini` model was $M_{\text{mini}} = 12.03$ s, compared with $M_{\text{nano}} = 9.53$ s for the `nano` model. A one-sided paired t-test (testing whether the mini model was slower than the nano model) confirmed that this difference was statistically significant ($t = 20.80$, $p < 0.001$). This means that the higher accuracy gains associated with the `mini` model (described in Section 6.4.1) came with a marked increase in response time.

Retrieval depth also produced a measurable latency effect. Providing the LLM with $k = 7$ retrieved chunks yielded a mean response time of $M_{k=7} = 10.98$ seconds. On the other hand, providing $k = 3$ retrieved chunks results in a mean response time of $M_{k=3} = 10.58$ seconds. Although the absolute difference was modest (approximately 0.4 seconds on average), a one-sided paired t-test (testing whether $k = 7$ was slower than $k = 3$) indicated that it was statistically significant ($t = 3.52$, $p < .001$). This means that increasing the number of retrieved chunks thus led to a small but measurable increase in latency.

Finally, latency varied substantially across retrieval approaches. A Friedman test revealed a significant overall effect of retrieval method on response time ($\chi^2(5) = 822.53$, $p <$

0.001). Mean latencies (in seconds) for each approach were: $M_{BM25} = 6.11$, $M_{OpenAISemantic} = 8.08$, $M_{OpenAIKeyword} = 8.56$, $M_{Vanilla} = 11.61$, $M_{GraphEager} = 14.97$, and $M_{GraphMMR} = 15.35$. Post-hoc Wilcoxon signed-rank tests with Holm correction showed that all pairwise differences were statistically significant ($p_{\text{corrected}} < 0.05$) except for the comparison between `OpenAI Keyword` and `OpenAI Semantic` (adjusted $p = 0.71$), whose latencies were statistically equivalent.

6.4.3. Cost

To complement the accuracy and latency analyses, we evaluated how each of the model size, retrieval depth, and retrieval method affected the monetary cost per request (Figure 5C). Costs were computed from input and output token usage under the October 2025 pricing for `gpt-5-mini-2025-08-07` and `gpt-5-nano-2025-08-07` [31].

Averaged across retrieval settings, the `mini` model was more expensive than the `nano` model. The `mini` variant cost $0.0028 per query on average, compared with $0.00056 for the texttnano model. A one-sided paired t-test confirmed that this difference was statistically significant ($t = 48.67$, $p < 0.001$), confirming that the additional power of the `mini` model comes with a substantially higher token cost.

Second, we assessed the impact of retrieval depth. Pipelines that used $k = 3$ retrieved chunks had a mean per-query cost of $0.0013, compared with $0.002 for $k = 7$. A one-sided paired t-test (testing whether $k = 7$ was more expensive than $k = 3$) indicated that this difference was statistically significant ($t = 18.25$, $p < 0.001$). Thus, on average, deeper retrieval nearly doubled the per-query cost, slightly increased the latency (Section 6.4.2) and produced modest gains in accuracy (Section 6.4.1).

Finally, we investigated cost differences across retrieval approaches, averaging over models and retrieval depths. A Friedman test showed a significant overall effect of retrieval method on cost ($\chi^2(5) = 942.05$, $p < 0.001$). Mean per-query costs were lowest for `Graph Eager` ($0.00064), followed by `Graph MMR` ($0.00097), `Vanilla` ($0.00168), `BM25` ($0.00185), `OpenAI Semantic` ($0.00240), and `OpenAI Keyword` ($0.00247). Post-hoc pairwise Wilcoxon signed-rank tests with Holm correction show that all methods differed significantly from one another (all adjusted $p < 0.05$).

Although the cost differences across methods were statistically significant, their practical impact is modest in our anticipated deployment setting. Even the most expensive configurations differ by only a few thousandths of a dollar per query. At an expected volume of fewer than 10,000 queries per month, these differences amount to only a few dollars in monthly spend. Thus, while statistically significant, the observed cost effects are not likely to be decisive for chatbot usage at our scale.

6.5. Best Pipeline Selection

Selecting a single RAG configuration for deployment requires balancing the competing objectives of accuracy, latency, and cost. Our experimental findings indicate that no single pipeline dominates across all three criteria, reinforcing the inherently multi-objective nature of RAG system optimization. Out of the 24 pipelines, the OpenAI keyword retrieval method paired with `gpt-5-mini-2025-08-07` consistently produced the highest accuracy scores, reaching 86.66% and 83.33%, respectively, when using retrieval depths of $k = 7$. The average latency and cost values for both pipelines were approximately 10 seconds and $0.005, respectively.

Subsequently, we followed the evaluation protocol described in Section 6.3 and conducted a blind human-expert assessment of the top-performing pipelines. Eleven manufacturing and occupational safety experts, including seven external to our research team, participated in the review. They evaluated responses from the two most accurate pipelines (`gpt-5-mini` paired with `OpenAI Keyword` and `OpenAI Semantic` retrieval at $k = 7$) using a randomly selected subset of ten benchmark questions. Across the resulting 110 blinded comparisons (11 experts × 10 questions), the experts preferred the the `OpenAI Keyword` retrieval in 45 (41%) cases, the `OpenAI Semantic` retrieval in 29 (26%) cases, and judged 36 (33%) cases as ties. Taken together, the expert evaluations corroborated the LLM-as-Judge evaluation approach and decisively supported the `gpt-5-mini-2025-08-07` + `OpenAI Keyword` + `top-k = 7` pipeline as the most suitable configuration for deployment. Although this configuration is not the cheapest nor the fastest, it offers the best trade-off for safety-critical environments. It maximizes retrieval and generative accuracy (the system's most important requirement), while maintaining acceptable levels of cost and latency.

7. Concluding Remarks

This study introduced a multimodal, retrieval-augmented chatbot framework designed to advance safety training and hazard recognition within Industry 5.0 manufacturing environments. Grounded in a rigorous Design Science Research (DSR) methodology, the work systematically addressed three critical requirements for next-generation safety systems, namely, accuracy, latency, and cost, through an experimentally validated approach. The final system integrates text, image, and voice modalities with multiple retrieval strategies to deliver contextually grounded, real-time safety guidance. Beyond the artifact itself, a secondary contribution of our work is a publicly available benchmark that can help to evaluate future AI solutions in manufacturing settings.

We argue that our empirical results are of great value to practitioners, as they demonstrate how different retrieval strategies and model configurations can exert a significant influence on system performance. In particular, our findings highlight the multi-objective nature of RAG configuration optimization, where trade-offs among correctness, speed, and expense must be balanced. This leads to interesting future research directions, where one can explore Pareto-based optimization frameworks to systematically identify non-dominated solutions that best balance accuracy, latency, and cost given a manufacturing facility's needs and budget. Our results with a local RAG approach suggest another promising avenue for future work. In particular, by fine-tuning lightweight local LLMs and combining them with efficient, on-premise RAG pipelines, one can potentially reduce both operational costs and response latency while preserving accuracy. These directions can further enhance our contribution of a framework and empirical foundation for next-generation AI-driven safety training systems, which marks a step toward human–AI collaboration that enhances safety, efficiency, and resilience in smart manufacturing environments.

Data and Code Availability

The code and evaluation benchmarks used in this study are publicly available at https://github.com/fmegahed/safety_rag_evaluation.

Acknowledgments

This work was funded by the Ohio Bureau of Workers' Compensation Worker Safety Innovation Center (WSIC) Grant #WSIC26-250206-009. Dr. Megahed and Carvalho also received support from the *Raymond E. Glos Professorship* and the *Dinesh & Ila Paliwal Innovation Chair* at Miami University. The authors acknowledge the *Farmer School of Business* for providing the computing infrastructure used to host our chatbot. The authors also appreciate the valuable feedback provided by Engineered Profiles, Yamaha Motor Company of North America, and MeetKai, Inc. on our benchmark Q&A datasets.

References

[1] Bureau of Economic Analysis, Gross domestic product 2nd quarter 2025 (third estimate), gdp by industry, corporate profits (revised), and annual update, https://www.bea.gov/sites/default/files/2025-09/gdp2q25-3rd.pdf (2025).

[2] U.S. Bureau of Economic Analysis, Table 6.4d. full-time and part-time employees by industry, https://apps.bea.gov/iTable/?reqid=19&step=2&isuri=1&1921=survey, accessed: October 28, 2025 (2025).

[3] National Center for Science and Engineering Statistics, Research and development: U.S. trends and international comparisons, https://ncses.nsf.gov/pubs/nsb20246/u-s-business-r-d, accessed: October 28, 2025 (2024).

[4] X. Xu, Y. Lu, B. Vogel-Heuser, L. Wang, Industry 4.0 and industry 5.0—inception, conception and perception, Journal of Manufacturing Systems 61 (2021) 530–535.

[5] Y. Lu, H. Zheng, S. Chand, W. Xia, Z. Liu, X. Xu, L. Wang, Z. Qin, J. Bao, Outlook on human-centric manufacturing towards industry 5.0, Journal of Manufacturing Systems 62 (2022) 612–627.

[6] U. B. of Labor Statistics, Employer-reported workplace injuries and illnesses – 2023, https://www.bls.gov/news.release/osh.htm, accessed: October 28, 2025 (2024).

[7] Liberty Mutual Insurance, Workplace Safety Index 2025, Annual report, Liberty Mutual Insurance, Boston, MA, retrieved from https://business.libertymutual.com/workplace-safety-index/ (2025).

[8] C. Peterson, L. Xu, S. B. L. Barnett, Average lost work productivity due to non-fatal injuries by type in the usa, Injury Prevention 27 (2) (2021) 111–117.

[9] K. S. Kiangala, Z. Wang, An experimental hybrid customized ai and generative ai chatbot human machine interface to improve a factory troubleshooting downtime in the context of industry 5.0, The International Journal of Advanced Manufacturing Technology 132 (5) (2024) 2715–2733.

[10] M. Mayyas, S. P. Vadlamudi, M. A. Syed, Fenceless obstacle avoidance method for efficient and safe human–robot collaboration in a shared work space, International Journal of Advanced Robotic Systems 17 (5) (2020) 1729881420959018.

[11] K. Peffers, T. Tuunanen, M. A. Rothenberger, S. Chatterjee, A design science research methodology for information systems research, Journal of Management Information Systems 24 (3) (2007) 45–77.

[12] X. Zhu, R. Y. M. Li, M. J. C. Crabbe, K. Sukpascharoen, Can a chatbot enhance hazard awareness in the construction industry?, Frontiers in Public Health 10 (2022) 993700.

[13] J. Jeon, Y. Sim, H. Lee, C. Han, D. Yun, E. Kim, S. L. Nagendra, M. B. Jun, Y. Kim, S. W. Lee, et al., ChatCNC: Conversational machine monitoring via large language model and real-time data retrieval augmented generation, Journal of Manufacturing Systems 79 (2025) 504–514.

[14] T. Wang, J. Fan, P. Zheng, An LLM-based vision and language cobot navigation approach for human-centric smart manufacturing, Journal of Manufacturing Systems 75 (2024) 299–305.

[15] S. Lou, R. Tan, Y. Zhou, Z. Zhao, Y. Zhang, C. Lv, Large language model-enabled cognitive agent for self-aware manufacturing, Journal of Manufacturing Systems 82 (2025) 1213–1226.

[16] H. Fan, C. Liu, N. E. Janvisloo, S. Bian, J. Y. H. Fuh, W. F. Lu, B. Li, MaViLa: Unlocking new potentials in smart manufacturing through vision language models, Journal of Manufacturing Systems 80 (2025) 258–271.

[17] S. Li, H.-L. Xie, P. Zheng, L. Wang, Industrial metaverse: A proactive human-robot collaboration perspective, Journal of Manufacturing Systems 76 (2024) 314–319.

[18] A. Gautam, M. R. Aryal, S. Deshpande, S. Padalkar, M. Nikolaenko, M. Tang, S. Anand, Iiot-enabled digital twin for legacy and smart factory machines with llm integration, Journal of Manufacturing Systems 80 (2025) 511–523.

[19] Office of Advocacy, Facts about small business: Manufacturing statistics 2025, U.S. Small Business Administration. https://advocacy.sba.gov/2025/03/10/facts-about-small-business-manufacturing-statistics-2025/, accessed on 2025-11-14 (Mar. 2025).

[20] B. Armstrong, B. Bonvillian, S. Berger, Advanced technology, advanced training: A new policy agenda for U.S. manufacturing, Tech. rep., MIT Initiative for Knowledge and Innovation in Manufacturing (May 2021).

[21] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive NLP tasks, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, Vol. 33, Curran Associates, Inc., 2020, pp. 9459–9474.

[22] Artifex, PyMuPDF documentation, https://pymupdf.readthedocs.io/en/latest/, accessed: 2025-11-14 (Nov. 2025).

[23] Mistral AI Team, Mistral OCR, https://mistral.ai/news/mistral-ocr, accessed on 2025-10-26 (Mar. 2025).

[24] OpenAI, Retrieval API, https://platform.openai.com/docs/guides/retrieval, accessed on 2025-10-26 (2025).

[25] LangChain, LangChain overview - docs by LangChain, https://docs.langchain.com/oss/python/langchain/overview, accessed: 2025-11-14 (2025).

[26] S. Robertson, H. Zaragoza, The probabilistic relevance framework: Bm25 and beyond, Foundations and Trends in Information Retrieval 3 (4) (2009) 333–389.

[27] LangChain, Graph RAG, https://python.langchain.com/docs/integrations/retrievers/graph_rag/, accessed on 2025-10-26 (2025).

[28] IBM, Datastax astra, https://accounts.datastax.com/session-service/v1/login, accessed: 2025-11-14 (2025).

[29] streamlit, streamlit v1.50.0, Python library. Available at https://pypi.org/project/streamlit/ (Sep. 2025).

[30] LangChain, Evaluate a RAG application, https://docs.langchain.com/langsmith/evaluate-rag-tutorial, accessed on 2025-10-27 (2025).

[31] OpenAI, Pricing, https://openai.com/api/pricing/, accessed on 2025-10-27 (Aug. 2025).