Trustworthy-by-Design Generative AI Assistants for Industrial Troubleshooting: A Knowledge Graph Grounded Architecture

Rohan Jadhav[1], Emmanuel Papadakis[1] and George Baryannis[1,*]

[1]School of Computing and Engineering, University of Huddersfield, Queensgate, Huddersfield, HD1 3DH, United Kingdom

Abstract

Manufacturing organisations increasingly face knowledge drain as specialised technical knowledge remains distributed across experienced staff, design documents, troubleshooting guides and historical correspondence. This paper presents an early industrial case study of a trustworthy-by-design generative AI architecture for troubleshooting in manufacturing, combining: human-validated knowledge graph construction, ontology-guided extraction, graph-based retrieval, multi-agent orchestration, query verification and source-oriented explanation. The paper shows how selected trustworthiness requirements, namely technical robustness and safety, transparency, and human agency and oversight, can be operationalised during system design. We discuss the industrial context, trustworthiness scope, architectural mechanisms and early implementation observations, including the role of retrieval strategy, document structure and human validation. The paper contributes a practical case experience showing how human-centred oversight, structured grounding and lifecycle-oriented design can support more trustworthy generative AI assistants in industrial settings.

Keywords

Trustworthy Artificial Intelligence (T-AI), Generative AI assistants, Knowledge graphs, Agentic AI, Human-in-the-loop validation, Industrial troubleshooting, Manufacturing

1. Introduction

Manufacturing organisations increasingly depend on specialised technical knowledge distributed across experienced staff, design documentation, troubleshooting guides, customer correspondence and internal procedures. This challenge is particularly acute for bespoke, designed-to-order products, where support and maintenance often require context-specific interpretation rather than retrieval of generic instructions. As experienced personnel retire, move roles, or become unavailable, organisations face a knowledge drain that can affect training, customer support and operational continuity [1].

Generative AI (GenAI) assistants offer a promising interface for making such knowledge more accessible to employees and customers. However, their use in industrial troubleshooting raises trustworthiness concerns beyond those of many consumer-facing chatbots. Incorrect or unsupported guidance may lead to inappropriate maintenance actions, operational disruption, avoidable cost, or safety-relevant consequences, especially when users lack the expertise to distinguish plausible from correct responses.

A central limitation of large language models (LLMs) is that their outputs are generated from statistical patterns rather than explicit, validated representations of organisational knowledge [2]. Retrieval-augmented generation partially addresses this by grounding responses in external documents, but document-level retrieval may be insufficient when troubleshooting depends on relationships between components, symptoms, causes, procedures and constraints. Knowledge graphs (KGs) offer a form of grounding by representing domain knowledge in a structured, queryable form [3].

This paper presents an early industrial case study of a trustworthy-by-design GenAI assistant for manufacturing troubleshooting. Rather than treating trustworthiness as a post-hoc evaluation

TRUST-AI: The Second European Workshop on Trustworthy AI. Organized as part of the International Joint Conference on Artificial Intelligence - IJCAI/ECAI 2026. August 2026, Bremen, Germany.

*Corresponding author.

r.jadhav@hud.ac.uk (R. Jadhav); e.papadakis@hud.ac.uk (E. Papadakis); g.bargiannis@hud.ac.uk (G. Baryannis)

0000-0001-8669-2420 (E. Papadakis); 0000-0002-2118-5812 (G. Baryannis)

concern, the architecture embeds mechanisms across the development lifecycle, including workflow standardisation, human-validated KG construction, KG-grounded retrieval, multi-agent orchestration, query verification, response checks, source-oriented explanation, and planned feedback mechanisms.

The paper is guided by three research questions: (1) how can selected trustworthy Artificial Intelligence (AI) requirements be operationalised in a KG-grounded, multi-agent assistant for industrial troubleshooting?; (2) what lessons emerge when embedding trustworthiness mechanisms before deployment?; and (3) what risks remain around knowledge extraction, retrieval completeness, transparency and user reliance? The main contributions are: (1) a case-based architecture for a KG-grounded, multi-agent assistant for industrial troubleshooting and knowledge support; (2) mapping architectural mechanisms to selected trustworthy AI requirements, focusing on technical robustness and safety, transparency, and human agency and oversight; (3) early lessons and residual risks concerning document extraction, KG construction and retrieval completeness.

The paper is structured as follows. Section 2 reviews related work on trustworthy AI, KG-grounded generation and agentic AI. Section 3 introduces the industrial case and trustworthiness scope. Section 4 presents the proposed architecture. Section 5 discusses trustworthiness mapping, early implementation observations and limitations. Section 6 concludes the paper and outlines future work.

2. Background and Related Work

2.1. Trustworthy AI as a Lifecycle Concern

The rapid development of LLM-driven systems has increased the need to examine their reliability, dependability and effects on users and organisational workflows [4, 5]. Prior work has highlighted risks such as hallucinations, untruthful outputs and sycophancy, which can undermine appropriate reliance on AI-generated outputs [6, 7]. In industrial settings, these concerns extend beyond answer quality to process fit, user oversight and the ability to correct system behaviour over time. They are also amplified in multi-stage and agentic systems, where an initial failure may propagate into subsequent planning, tool use and response generation.

The European Commission's Assessment List for Trustworthy Artificial Intelligence (ALTAI) provides a useful lens for considering trustworthiness across the lifecycle of an AI system [8]. Its requirements include human agency and oversight, technical robustness and safety, privacy and data governance, transparency, diversity, non-discrimination and fairness, societal and environmental well-being, and accountability. This paper focuses on the requirements most directly relevant to the current industrial case: technical robustness and safety, transparency, and human agency and oversight. Trust is also dynamic: Regona et al. [9] emphasise that it depends on communication, feedback collection and transparent updates over time. Trustworthiness therefore requires mechanisms for validation, monitoring, correction and controlled evolution [9].

2.2. Knowledge Graph Grounding and Quality

Retrieval-augmented generation has become a common approach for reducing unsupported LLM outputs by grounding generation in external sources [10]. However, retrieval over unstructured documents can struggle when relevant information is distributed across sections, diagrams, tables, component descriptions and procedural dependencies, particularly in technical industrial documents [11]. In such cases, the issue is not only whether a relevant document is retrieved, but whether the system retrieves and reasons over the correct relationships between domain entities.

KGs provide a structured representation of entities, relationships and properties, making them useful where traceability and relational context are important. Graph-based retrieval can support retrieval over structured relational context rather than isolated text chunks [12]. Prior work has shown that KG-grounded chatbots can improve factuality and user satisfaction in question-answering scenarios [13], while domain-specific KGs have also been used to reduce hallucination and improve transparency in

specialised applications [14]. In this paper, the KG is used not as a guarantee of correctness, but as a mechanism for structured grounding, retrieval control and traceability.

KG grounding also introduces its own trustworthiness challenges. A KG can only support trustworthy responses if its contents are accurate, sufficiently complete and appropriately maintained. Relevant quality dimensions include accuracy, completeness, consistency, timeliness and redundancy [15]. These are difficult to manage when KGs are populated from heterogeneous industrial documents such as manuals, troubleshooting guides, diagrams, tables and correspondence. Moreover, LLM- or vision-language-model-based extraction can introduce incorrect associations or hallucinated content. This makes human-in-the-loop validation central to industrial KG construction, since domain experts can judge whether extracted information is technically meaningful and correctly represented [15, 16].

2.3. Agentic AI and Trustworthy Orchestration

Agentic AI systems decompose tasks across specialised components that may plan, retrieve information, verify inputs, execute actions or generate responses. This decomposition can support trustworthiness by reducing the scope of each component and enabling more explicit control over system behaviour. Vandeputte [17] argues that robust GenAI-native systems should be designed for fault tolerance and transparency of processing. Prior work on conversational agents has also shown that controllability, transparency and clear feedback can improve user trust in chatbot interactions [18].

In industrial troubleshooting, multi-agent orchestration is useful because user queries may combine multiple intents, require clarification, depend on domain-specific tools, or involve several possible hypotheses. Research on LLM-based agents has shown how language models can act as controllers that plan tasks, select specialist tools or models, execute subtasks and summarise outputs [19]. Query verification and action planning are also relevant where user-provided information may be incomplete, ambiguous or inconsistent with available domain knowledge [20].

However, agentic decomposition also creates trustworthiness risks: failures may arise from query interpretation, incorrect assumptions, incomplete retrieval or inappropriate tool invocation. A trustworthy design must therefore enable inspection and governance [5]. This motivates architectural mechanisms such as scoped agents, human validation, source-oriented explanation and feedback mechanisms. The architecture described in Section 4 builds on this view by treating agentic orchestration, KG grounding and human validation as complementary trust mechanisms.

3. Industrial Case Context

3.1. Industrial Setting

The case concerns the development of a GenAI assistant for an industrial manufacturing organisation producing bespoke, designed-to-order products. The organisation's technical knowledge is distributed across design documentation, troubleshooting guides, manuals, diagrams, tables, historical correspondence and expert personnel. This creates a practical knowledge management challenge: employees and customers may require accurate support, but the relevant knowledge may be dispersed, implicit, difficult to search, or dependent on experienced staff.

The assistant is intended to support two broad user groups. First, it acts as a knowledge support tool for employees, especially less experienced staff who need guidance on product knowledge and internal procedures. Second, it is intended to support customers seeking technical information or troubleshooting guidance. In both cases, users may not always be able to judge whether a generated answer is correct or complete, making trustworthiness a central design requirement.

The system is being developed from heterogeneous industrial knowledge sources, including technical documents, design manuals, troubleshooting material, diagrams, tables and correspondence. Much of this material is stored in PDF format, with varying levels of structure and quality. The development process therefore requires document preprocessing, extraction of entities and properties, ontology construction, KG population and human validation.

This setting differs from many open-domain chatbot applications because the relevant knowledge is proprietary, domain-specific and operationally important. It is also not sufficient to retrieve text snippets alone. Troubleshooting often depends on relationships between symptoms, components, procedures, assumptions and constraints. The architecture therefore uses a KG as a structured knowledge layer, combined with a multi-agent assistant that can route queries, plan retrieval, verify query information, explore hypotheses and generate responses.

3.2. Trustworthiness Scope

Drawing on ALTAI [8], this paper focuses on the trustworthiness requirements that are most directly affected by the design and implementation of the proposed AI-assisted industrial system. Specifically, our scope is limited to technical robustness and safety, transparency, and human agency and oversight, as these requirements are directly influenced by the system architecture, human–AI interaction, and operational deployment decisions presented in this work. These aspects can therefore be meaningfully assessed through the implemented architecture and its validation.

Other requirements such as privacy and accountability are recognised as important for trustworthy AI but are outside the primary scope of this study. Accountability depends on governance, auditing, and organisational oversight beyond system design. Privacy involves broader lifecycle controls such as consent, access management, and data governance, especially in graph-based systems. The current research outcomes mainly support robustness, transparency and human agency to improve trustwortiness of an AI Assistant, while other requirements remain a focus for future study.

Within this scope, the KG functions as a structured grounding mechanism that can support trustworthiness when combined with ontology constraints, source-aware extraction, retrieval controls and human validation. Table 1 summarises how the selected requirements are interpreted in the industrial case. The trustworthiness scope defined in this section motivates the architecture presented next.

Table 1
Trustworthiness scope of the industrial case

Requirement	Risk in the case	Design response
Technical robustness and safety	Unsupported or incomplete troubleshooting guidance due to extraction, retrieval or generation errors.	Ontology-guided KG construction, KG-grounded retrieval, query verification and human validation.
Transparency	Users and developers need to distinguish data-backed content from assumptions or synthesis.	Explicit entities, properties and relationships, source metadata and evidence surfacing.
Human agency and oversight	Domain expertise is needed to judge whether extracted knowledge is complete and technically meaningful.	Editable KG records and human-in-the-loop validation of nodes, properties and relationships.

4. Trustworthy-by-Design Architecture

To instantiate the requirements defined in Section 3, we propose an end-to-end architecture anchored by a Knowledge Graph. The architecture has two components: KG construction and question answering. The first combines ontology-guided extraction with human-in-the-loop validation to create a curated knowledge layer. The second uses a specialised multi-agent ecosystem to ground troubleshooting responses, verify user premises and reduce unsupported generation.

4.1. Knowledge Graph Construction

Relevant manufacturing knowledge was stored in unstructured PDFs, including manuals, troubleshooting documents, schematics, tables, figures and charts. These documents were converted into a structured representation before ontology-guided extraction and KG population. Figure 1 summarises the workflow.

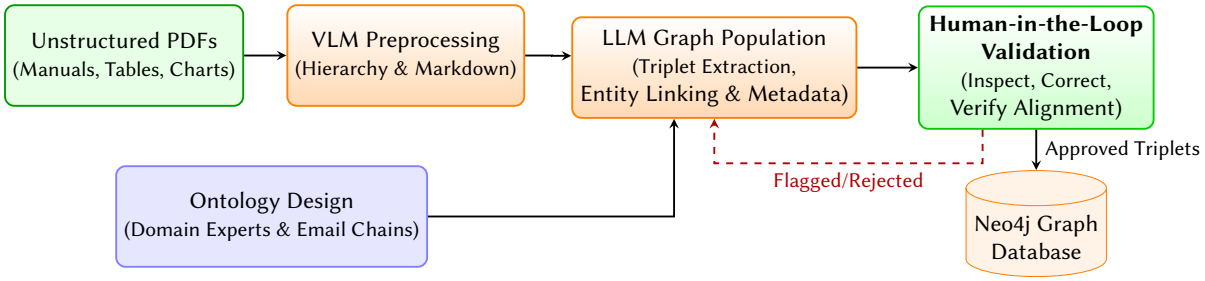

Figure 1: Workflow of knowledge graph construction featuring human-in-the-loop validation.

4.1.1. Preprocessing

Vision-Language Models (VLMs) were used for document preprocessing because empirical tests showed that they handled document structure better than traditional PDF and OCR tools such as PyMuPDF, pdfplumber, EasyOCR and Tesseract. These tools often flattened headers, footers, captions, table fragments and image-related text into the main body, reducing suitability for KG construction. Through prompting, VLMs were instructed to remove structural noise, preserve hierarchy, associate captions with visual elements and extract text in markdown format. This preserved contextual cues from headings, sections, tables and figures, which later supported KG node and property grounding.

4.1.2. Ontology design

An ontology was designed to constrain extraction through an explicit manufacturing domain model. Initial concepts were identified from competency questions derived from customer-service examples, design documentation, troubleshooting material and consultation with domain experts. These were then organised into entity types, relationships and data properties, defining which nodes could be created, which relationships were meaningful and which properties applied to each entity type.

LLM-based ontology generation was explored but not adopted as the primary strategy. Although automated approaches can support concept, relationship and property extraction [21], in our setting LLM-generated ontologies were incomplete and structurally weaker than the manually constructed version. The ontology was therefore treated as a human-guided artefact.

4.1.3. Ontology-guided graph population

To populate the KG, LLMs are tasked with extracting candidate triplets, constrained by the domain-specific ontology, from preprocessed documents. Additional metadata was manually added where needed, including document visibility level (e.g., private, public, or customer-specific), relevant solver type, image assessment rules, and source information. This metadata is important because the assistant may need to determine whether a piece of knowledge is appropriate for a specific user, use case, or reasoning workflow.

The risk of LLM attention drift was mitigated through strategic document chunking, in order to preserve the structural and contextual integrity of the content. A common risk of chunking is fragmented knowledge. To mitigate this, we applied entity normalisation, where node names are constrained to short, lowercase, type-specific labels, and candidate nodes with matching normalised names are merged

with their properties consolidated. This reduces duplicate representations and supports a more coherent graph structure.

4.1.4. Human-in-the-loop validation and quality control

Because LLM-based extraction can produce hallucinated, incomplete, duplicated, or incorrectly structured triples, candidate triples are validated before being stored in the graph database. Domain experts inspect the extracted triples, check their alignment with the ontology, and verify whether the entities, relations, and properties are supported by the source documents. Unsupported or ambiguous triples are corrected, rejected, or flagged for further review.

After validation, the approved triples are stored in Neo4j and made available to the multi-agent question-answering layer. This ensures that downstream retrieval and response generation are grounded in curated knowledge rather than directly in unverified LLM outputs.

4.2. Question answering

The second component of the architecture is a KG-grounded question answering ecosystem. This system uses the KG as the main source of domain evidence for troubleshooting responses; hence it constrains response generation through retrieved knowledge from the industrial document and ontological constraints.

4.2.1. System orchestration

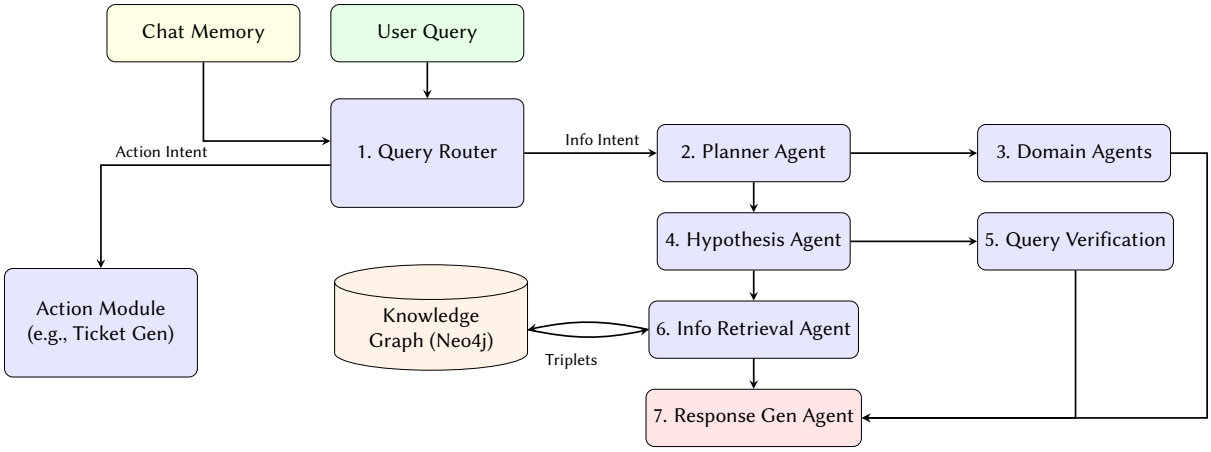

Figure 2: The 7-Agent architecture guided by knowledge graph rules.

The ecosystem is organised into a loosely coupled seven-agent topology. These agents are governed by a central orchestrator that operates based on the workflow depicted in Figure 2. The agents can exchange messages that contain the generated response, the current agent identifier, and the conversation history. The reasoning of this entire ecosystem is strictly governed by the constraints and rules defined within the KG ontology. The workflow is organised around two high-level paths: an action path and a response generation path. When a user submits a query, an initial routing agent analyses the intent and determines whether the request corresponds to a predefined action, such as ticket generation or document search, or whether it requires an information-seeking response grounded in the knowledge base. This routing step limits unsupported generation by preventing the assistant from attempting tasks outside the implemented system capabilities.

Beyond the immediate request, the routing agent also draws on session memory to inform its routing decisions, maintaining a running record of the conversation's context rather than treating each request in isolation. This includes previous commands the user has issued, which give the agent visibility into

the ongoing workflow; inferred intents detected from earlier turns, which preserve continuity even when a new request does not restate them explicitly; prior queries, which help the agent recognise follow-ups, clarifications, or corrections; and past responses, which help it avoid redundant routing and account for what the user has already received. By incorporating this historical context, the routing agent can resolve ambiguous references, maintain continuity across multi-turn interactions, and route each request based on the full conversational trajectory. The result is routing that is more accurate, better aligned with user intent throughout the session, and better supports consistent decision-making.

4.2.2. Solution seekers

If the intent requires troubleshooting support, a planner agent formulates an execution strategy to coordinate the subsequent workflow. When specific operational logic or calculations are required, domain agents handle these tasks utilising predefined solvers mapped to the domain; for example predefined actions include executing programmatic searches for specific schematics, python scripts or other industrial tools and services that are invoked using the user's input.

The analytical depth of the system relies on the structured constraints of the graph. When a user presents an ambiguous problem or a failure with multiple potential root causes, the planner agent activates a hypothesis generation agent to explore scenarios. This is achieved by navigating KG links that connect specific machinery components to known faults, observed symptoms, and documented solutions. This ensures that all proposed hypotheses are mechanically valid. Concurrently, a query verification agent prevents the system from accepting flawed user premises by cross-referencing the user's initial assumptions directly against the deterministic entities and properties stored in the graph.

4.2.3. Information retrieval and response generation

Access to the knowledge repository is managed exclusively by the information retrieval agent, which translates query parameters originated by other agents into targeted search operations. Once the necessary sub-graphs are extracted, a response generation agent synthesises the final answer. The KG and its underlying ontology are fundamental to this generation phase. The response agent does not generate text unconstrained; rather, it structures its natural language output strictly around the retrieved triplets, properties, and metadata. By citing the underlying graph entities used in its reasoning, the response generation agent guarantees that the output remains entirely grounded in verified industrial rules, ensuring a transparent and auditable workflow.

4.2.4. Retrieval strategies

While the architecture requires all agents to function cohesively, the factual accuracy of the final response is fundamentally bounded by the quality and completeness of the retrieved data. Consequently, the design and validation of the system heavily prioritise the information retrieval agent.

This agent utilises two distinct retrieval strategies: a) The similarity strategy stores embeddings of the graph triplets and data properties in a vector database, returning the top matching nodes based purely on semantic distance; b) The hopping strategy extends this baseline by identifying initial semantic seed nodes via vector similarity, and subsequently traversing the explicit graph edges to a predefined depth (e.g., 2, 4, or 6 hops). This traversal allows the system to uncover latent relational constraints and interconnected components that a simple semantic search would miss.

The aforementioned strategies were measured based on initial sample 20 queries asked by an domain expert to the chatbot, with results in Table 2. Among these, the 4-hop strategy performed better compared to other strategies in terms of completeness and noise trade-off. The measured average latency suggests that retrieval time remains relatively stable for AI Assistants.

Metric	**2-hop**	**4-hop**	**6-hop**	**Similarity**
Accuracy	15/20	17/20	16/20	12/20

Metric	2-hop	4-hop	6-hop	Similarity
Retrieval Latency (avg, sec)	0.77	0.71	0.77	0.33

Table 2: Comparison of retrieval strategies across accuracy and latency metrics.

5. Discussion

The case study shows how trustworthiness can be approached as a design concern before full deployment, rather than only as a post-hoc evaluation problem. The architecture operationalises the selected ALTAI requirements mainly through three mechanisms. First, technical robustness and safety are supported by ontology-guided KG construction, KG-grounded retrieval, query verification and human validation. Second, transparency is supported by representing knowledge as explicit entities, properties and relationships, allowing responses to be connected to retrieved graph content and source metadata. Third, human agency and oversight are supported by treating the KG as an editable knowledge record that domain experts can inspect, correct and extend without retraining the underlying LLM.

The implementation remains at an early stage, but already provides useful evidence about where trustworthiness risks arise. The initial KG was populated from selected design documentation and currently contains 1033 nodes and 1059 edges, with no isolated nodes or self-loops. In a preliminary evaluation using 20 confidential expert queries, the 4-hop retrieval strategy produced the highest number of correct responses, with 17/20 responses manually validated as correct against the source design manuals. Partial, factually incorrect or empty responses were treated as incorrect. This suggests that graph traversal can improve retrieval completeness compared with simple similarity retrieval, but also shows that retrieval depth must be controlled. In some cases, deeper traversal introduced additional context that appeared to add noise rather than improve answer quality.

For complex queries, the chatbot's generated responses using a 4-hop retrieval strategy were further evaluated by domain experts. Feedback was collected using three metrics: correctness, completeness, and relevance, along with an optional free-text feedback field. The results, presented in Table 3, show that all incorrect queries were also marked as incomplete, indicating a need for improved retrieval. The experts' textual feedback for incorrect and incomplete responses highlighted missing items, suggesting that the retriever was not able to capture all relevant categories or examples for a given query. They also noted the need for the chatbot to clarify or further explore ambiguous queries before generating a response, as well as instances of chatbot not remembering historical context for generating response.

Metric	100%	0%	In-between	Weighted Average
Correctness	48	15	8	73.56
Completeness	47	16	8	70.96
Relevance	50	12	9	76.24

Table 3: Evaluation breakdown across correctness, completeness, and relevance.

Several lessons emerge from the case study. First, document quality is a trustworthiness issue, not merely a preprocessing issue. Industrial PDFs may contain tables, figures, charts and captions whose role is not explicit, and visual artefacts may contain technical knowledge that cannot be recovered reliably through text extraction alone. Second, KG grounding is only as reliable as the extraction and validation process used to construct the graph. LLM- and VLM-based extraction can introduce hallucinated entities, missing properties or incorrect associations, so human-in-the-loop validation remains essential. Third, retrieval is a key trustworthiness bottleneck. Even where relevant information exists in the KG, the assistant may fail if the retrieval strategy does not reach the appropriate nodes or retrieves excessive neighbouring information.

The case study also highlights limitations that need to be addressed before wider deployment. The current KG covers only part of the available documentation, so the findings should be interpreted as

early implementation observations rather than a complete evaluation. The current entity normalisation approach, based on short type-specific labels and node merging, helps reduce duplicate nodes but does not yet provide full co-reference resolution or ontology-level entity alignment. This may affect cross-section connectivity and retrieval quality as the graph expands. Similarly, the current validation process provides fine-grained control over KG quality but is labour-intensive. A more scalable approach would use expert corrections as feedback signals, prioritising triples involving low-confidence extractions, safety-relevant parameters, unresolved duplicates or ontology violations for review.

Overall, the case supports the paper's central argument that trustworthy industrial GenAI assistants require architectural controls across the development lifecycle. KG grounding, agentic orchestration and human validation provide complementary mechanisms, but none is sufficient alone. Future iterations must therefore strengthen retrieval, provenance tracking, evidence surfacing, entity alignment and feedback-driven validation before the assistant can be assessed as a mature trustworthy AI system.

6. Conclusion

This paper presented an early industrial case study of a trustworthy-by-design GenAI assistant for manufacturing troubleshooting. It showed how human-validated KGs, ontology-guided extraction, graph-based retrieval and multi-agent orchestration can support technical robustness, transparency, and human agency and oversight. Future work will improve retrieval, expand KG coverage, handle implicit rules and visual artefacts, and evaluate whether explanations are useful and actionable for different user groups, drawing on human feedback on user experience, perceived usefulness, explanation quality and appropriate reliance [22].

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT and Gemini for: Grammar and spelling check, Paraphrase and reword. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

Acknowledgments

This work was supported by Innovate UK [project reference 10151759].

References

[1] N. Galan, Knowledge loss induced by organizational member turnover : a review of empirical literature, synthesis and future research directions (Part I), Learning Organization 30 (2023) 117–136. doi:10.1108/tlo-09-2022-0107, cC BY 4.0.

[2] C. Wang, X. Liu, Y. Yue, Q. Guo, X. Hu, X. Tang, T. Zhang, C. Jiayang, Y. Yao, X. Hu, Z. Qi, W. Gao, Y. Wang, L. Yang, J. Wang, X. Xie, Z. Zhang, Y. Zhang, Survey on Factuality in Large Language Models, ACM Comput. Surv. 58 (2025). doi:10.1145/3742420.

[3] X. Guan, Y. Liu, H. Lin, Y. Lu, B. He, X. Han, L. Sun, Mitigating large language model hallucinations via autonomous knowledge graph-based retrofitting, in: Proc. AAAI Conference on Artificial Intelligence, volume 38, 2024, pp. 18126–18134.

[4] L. D. Thomas, A. K. G. Romasanta, L. Pujol Priego, Jagged competencies: Measuring the reliability of generative AI in academic research, Journal of Business Research 203 (2026) 115804. doi:10.1016/j.jbusres.2025.115804.

[5] E. Barbu, M. Domnich, R. Vicente, N. Sakkas, A. Morim, Exploring Commonalities in Explanation Frameworks: A Multi-Domain Survey Analysis, 2025. arXiv:2405.11958.

[6] M. Sharma, M. Tong, T. Korbak, D. Duvenaud, A. Askell, S. R. Bowman, N. Cheng, E. Durmus, Z. Hatfield-Dodds, S. R. Johnston, S. Kravec, T. Maxwell, S. McCandlish, K. Ndousse, O. Rausch, N. Schiefer, D. Yan, M. Zhang, E. Perez, Towards Understanding Sycophancy in Language Models, 2025. arXiv:2310.13548.

[7] S. Lin, J. Hilton, O. Evans, TruthfulQA: Measuring How Models Mimic Human Falsehoods, in: S. Muresan, P. Nakov, A. Villavicencio (Eds.), Proc. 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 3214–3252. doi:10.18653/v1/2022.acl-long.229.

[8] High-Level Expert Group on Artificial Intelligence, Assessment List for Trustworthy Artificial Intelligence (ALTAI) for self-assessment, Technical Report, European Commission, 2020. URL: https://digital-strategy.ec.europa.eu/en/library/assessment-list-trustworthy-artificial-intelligence-altai-self-assessment, accessed: 2026-04-20.

[9] M. Regona, T. Yigitcanlar, C. Hon, M. Teo, Building Trust in Artificial Intelligence: A Systematic Review through the Lens of Trust Theory, ACM Comput. Surv. 58 (2026). doi:10.1145/3789256.

[10] 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: Proc. 34th International Conference on Neural Information Processing Systems, NIPS '20, Red Hook, NY, USA, 2020, pp. 9459 – 9474.

[11] H. Scaffidi, M. Hodkiewicz, C. Woods, N. Roocke, GraphRAG on Technical Documents - Impact of Knowledge Graph Schema, Transactions on Graph Data and Knowledge 3 (2025) 3:1–3:24. doi:10.4230/TGDK.3.2.3.

[12] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, D. Metropolitansky, R. O. Ness, J. Larson, From Local to Global: A Graph RAG Approach to Query-Focused Summarization, 2025. arXiv:2404.16130.

[13] P. Patel, Graph-Enhanced Retrieval-Augmented Question Answering for E-Commerce Customer Support, International Journal of Complexity in Applied Science and Technology 2 (2026) 200–211.

[14] T. Xu, J. Feng, J. Melendez, K. Roberts, D. Cai, M. Zhu, D. Elbert, Y. Chen, R. J. Bateman, Addressing accuracy and hallucination of LLMs in Alzheimer's disease research through knowledge graphs, 2025. arXiv:2508.21238.

[15] B. Xue, L. Zou, Knowledge Graph Quality Management: A Comprehensive Survey, IEEE Transactions on Knowledge and Data Engineering 35 (2023) 4969–4988. doi:10.1109/TKDE.2022.3150080.

[16] S. Tsaneva, D. Dessì, F. Osborne, M. Sabou, Knowledge graph validation by integrating LLMs and human-in-the-loop, Information Processing & Management 62 (2025) 104145. doi:10.1016/j.ipm.2025.104145.

[17] F. Vandeputte, Foundational Design Principles and Patterns for Building Robust and Adaptive GenAI-Native Systems, in: Proc. 2025 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software, Onward! '25, ACM, New York, NY, USA, 2025, p. 44–62. doi:10.1145/3759429.3762620.

[18] Y. Guo, J. Wang, R. Wu, Z. Li, L. Sun, Designing for trust: a set of design principles to increase trust in chatbot, CCF Transactions on Pervasive Computing and Interaction 4 (2022) 474–481. doi:10.1007/s42486-022-00106-5.

[19] Y. Shen, K. Song, X. Tan, D. Li, W. Lu, Y. Zhuang, HuggingGPT: solving AI tasks with chatgpt and its friends in hugging face, in: Proc. 37th International Conference on Neural Information Processing Systems, NIPS '23, Red Hook, NY, USA, 2023, pp. 38154 – 38180.

[20] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. R. Narasimhan, Y. Cao, ReAct: Synergizing Reasoning and Acting in Language Models, in: The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, 2023. URL: https://arxiv.org/abs/2210.03629.

[21] M. S. Abolhasani, R. Pan, Leveraging LLM for Automated Ontology Extraction and Knowledge Graph Generation, 2024. arXiv:2412.00608.

[22] N. Sakkas, S. Yfanti, P. Shah, N. Sakkas, C. Chaniotakis, C. Daskalakis, E. Barbu, M. Domnich, Explainable Approaches for Forecasting Building Electricity Consumption, Energies 16 (2023).