Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 00 (2025) 000–000

www.elsevier.com/locate/procedia

13th CIRP Global Web Conference (CIRPe 2025)

# Agentic Data Analysis for Intelligent Manufacturing: Benchmark-Driven Evaluation of Agentic vs. Direct LLM Approaches

Nastaran Moradzadeh Farid[a,*] , Alireza Taghizadeh[b], Sara Shafiee[a]

*[a]Technical University of Denmark, Department of Civil and Mechanical Engineering, 2800 Kgs. Lyngby, Denmark*
*[b]Configit A/S, Midtermolen 3, 4, 2100 Copenhagen, Denmark*

* Corresponding author. Tel.: +45 91727682. *E-mail address:* nmofa@dtu.dk

## Abstract

Recently, agentic artificial Intelligence (AI) has gained strong attention, showing promising results in domains such as quality control, knowledge management, and cost optimization. Yet, its application within manufacturing remains largely underexplored. Moreover, data analysis is a critical step across many manufacturing processes, but interpreting data often depends heavily on the expertise of technical professionals. To address this gap, this paper introduces a lightweight, agentic framework that enables non-expert users to interact with manufacturing datasets through natural language. The system integrates language models (LLMs) with modular tool orchestration to support data querying, analysis, and visualization via conversational interfaces. The system is evaluated using two representative manufacturing datasets and a benchmark of structured natural language queries inspired by TableBench. Comparative results across multiple LLMs reveal that our agentic approach outperforms direct prompting methods in both accuracy and interpretability. These findings demonstrate the feasibility and effectiveness of deploying agentic AI systems for real-world industrial data analysis and point toward more accessible and scalable AI-driven manufacturing solutions.

*Keywords:* Manufacturing, Data Analysis, AI, LLM, Agents, and LangGraph.

## 1. Introduction

Large Language Models (LLMs) and the agents built upon them have recently attracted significant attention across various domains. These agents extend the capabilities of LLMs by integrating them with external tools [1], enabling interaction with the environment, such as retrieving data from databases or executing specific actions [2]. Additionally, agents can access diverse knowledge-bases [3] and support human interaction through a concept known as *human-in-the-loop*, which allows humans to guide or control the decision-making process, thereby enhancing reliability [4]. Another important aspect of these agents is their orchestration and memory management capabilities, which empower them to perform complex reasoning and handle long, sequential tasks effectively.

Agentic AI systems have been successfully applied across a range of domains, including software engineering [5,6], web navigation [7,8], scientific research such as biology [9] and Machine Learning (ML) tasks [10], and conversational interfaces [11,12]. These applications demonstrate the agents' ability to automate complex reasoning processes and enhance user interaction.

An increasing body of research has investigated how agentic workflows can support data analysis and data science tasks. These approaches typically integrate LLMs with structured planning, reasoning, and error handling, enabling interaction with a central sandbox environment to execute tasks using tools such as Python and SQL interpreters. These systems enable automatic data transformation, exploratory data analysis, and insight generation. Recent reviews [13] categorize such agents into two broad types: conversational (multi-turn dialogue) and end-to-end (single-prompt execution). Conversational methods rely on interactive, multi-turn user input, allowing the agent to refine its responses through dialogue, such as MLCopilot [14] and LAMBDA [15]. In contrast, end-to-end approaches use a single prompt, enabling the agent to independently plan and execute tasks without further user intervention, such as Data Copilot [16] and AutoM3L [17].

In manufacturing, some recent studies have explored agentic AI applications. Gautam et al. [18] present an IIoT-enabled digital twin with a multi-agent LLM framework for real-time fault diagnosis, while Jeon et al. [19] propose ChatCNC, a conversational system linking CNC machines and manufacturing databases. Both works demonstrate the potential of LLM agents but remain limited in scope and evaluation scale. Recent work further shows how LLMs can enhance human–robot collaboration in assembly [20], while surveys of autonomous AI agents highlight their broader role in enabling reasoning and workflow orchestration across industrial domains [21]. Complementing this, deep learning has already proven effective for defect detection, predictive maintenance, and process optimization in manufacturing, providing a strong foundation for LLM-based methods [22]. In addition, hybrid retrieval-augmented generation frameworks have been introduced to adapt LLMs for domain-specific questions and answers in smart manufacturing [23].

Compared to these system-focused demonstrations, our work addresses a different aspect of the problem: enabling systematic, benchmark-driven evaluation of agentic AI for general manufacturing data analysis. Manufacturing processes can produce vast amounts of complex data from sensors, machines, and production systems [24]. As the manufacturing domain is often described as "data-rich but information-poor" [25], analyzing the data typically requires both domain expertise and proficiency in programming or data science, creating a challenging bottleneck for many industry practitioners [26]. Without accessible analytical tools, critical insights may be overlooked, slowing innovation and reducing operational efficiency. This challenge highlights the need for intuitive, human-centered AI systems capable of supporting complex data analysis through natural language interaction.

In this work, a system is proposed that leverages the capabilities of modern LLMs in combination with appropriate tools to enable natural language querying, interpretation, and visualization of structured datasets. Built on LangGraph, the proposed system facilitates the construction of agent-based workflows powered by LLMs, supporting modular reasoning and dynamic tool integration. While the framework is generalizable and can be applied to data analysis tasks across various domains, this paper focuses specifically on addressing the data analysis challenges associated with manufacturing area. The key contributions of this paper are:

- We propose and develop a simple, yet complete modular agentic AI system based on the LangGraph framework, specifically designed for manufacturing data analysis. The system enables natural language interaction with structured datasets, balancing simplicity, flexibility, avoiding unnecessary complexity while remaining effective. In practice, the system can support data analysis in key manufacturing tasks such as predictive maintenance, quality assessment, and process optimization.
- We design a structured evaluation approach tailored to the selected manufacturing datasets extending the TableBench [27] framework. This benchmark comprises 200 natural language questions with validated reference answers across two representative manufacturing datasets, enabling systematic performance assessment.
- We investigate the performance and effectiveness of agentic data analysis in the manufacturing domain by developing a small, structured benchmark.
- We compare the performance of various LLMs as the core reasoning engine within the system, assessing their performance in terms of accuracy, response time, and token usage. Our results demonstrate that the proposed system outperforms direct LLM prompting in data analysis tasks.

## 2. Method and Framework

This section outlines the design and development of the proposed agentic system, aimed at addressing the challenges of data analysis in the manufacturing domain. In this work, we investigate the feasibility and effectiveness of applying agentic AI to manufacturing data analysis. We design and implement an agentic system using the LangGraph framework, which builds based on LangChain with graph-based orchestration for step-by-step reasoning and dynamic tool use. We chose LangGraph because it offers a controllable and modular execution flow, making it more transparent and reproducible than CrewAI or OpenAI Swarm, and lighter and more flexible than AutoGen for prototyping [28]. To evaluate the system, we selected two representative manufacturing datasets: the Predictive Maintenance dataset (AI4I2020) [29] and the Intelligent Manufacturing Dataset (Man6GData) [30]. Drawing inspiration from the TableBench benchmark [27], we developed a structured set of natural language questions paired with reference answers to assess the proposed system's performance across a variety of scenarios. All appendices and data supporting this study are available at: https://github.com/Nastaran95/agentic-man-da.

### 2.1. Problem Formulation

The goal of the system is to empower non-expert users to interact with manufacturing datasets via natural language, removing the need for programming or advanced analytical skills. The system interprets user intent, identifies relevant data within structured tables, executes appropriate analytical operations, and returns results in a clear, conversational format. When applicable, the system also generates visualizations, such as plots or charts, to support data interpretation and enhance user understanding.

### 2.2. System Architecture

The system is structured around a modular, conversational architecture (Figure 1). The system operates as a conversational data analysis agent, tailored for natural language interaction with manufacturing data. It features a user interface that supports data management tasks, such as file uploads, user-

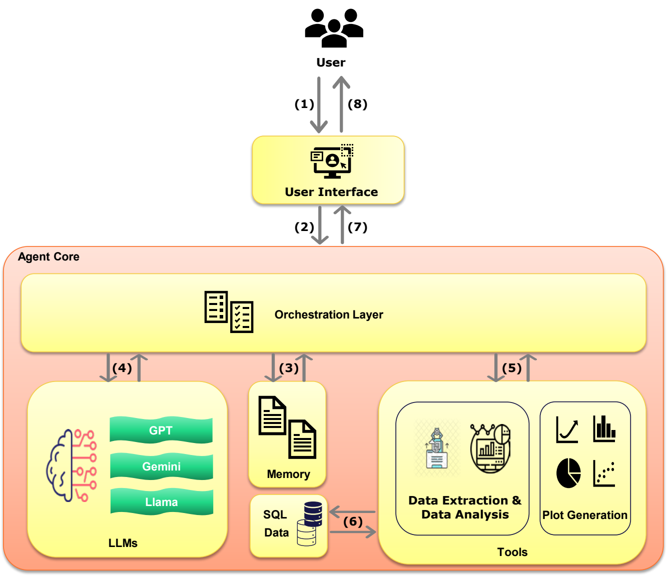

Figure 1 The proposed agentic system architecture. The flow includes: (1) user input queries via the user interface, (2) transfer of the query to the orchestration layer, (3) conversational memory management, (4) LLM calls, (5) tools usage (including data extraction interacting with (6) SQL and datasets, as well as plot generation), (7) output generation and visualization, and (8) returning the results to the user.

provided data descriptions, and previewing datasets, as well as a natural language interface for question-and-answer exchanges.

As shown in Figure 1, the system is composed of four core components including the user interface, the orchestration layer, analytical tools, and the underlying LLM. The user interface enables users to upload datasets, provide contextual descriptions, and pose natural language queries. The orchestration layer implemented using the LangGraph framework, governs the flow of information and decision-making across the system, mapping queries to tools, sequencing multi-step reasoning, and coordinating conversational state. Analytical modules (e.g., for SQL queries, analytical operations, or visualizations) are dynamically called based on user intent, executing domain-specific operations and returning structured outputs for integration. At the core, the LLM interprets natural language inputs, resolves ambiguities, and generates context-aware responses.

The performance of the system is evaluated using different LLMs, comparing their effectiveness in terms of accuracy and overall system behaviour. The system also maintains a memory of user interactions, using a lightweight structure that stores recent exchanges. At each turn, selected parts of recent history are appended to the LLM's input, allowing the agent to refine or expand previous answers and support multi-turn dialogue. This design not only enhances the continuity of the conversation but also supports more complex, multi-step reasoning processes. Our modular and agentic design ensures flexibility, scalability, and robustness, making the system suitable for real-world data-analysis in manufacturing environments.

## 3. Experiments

To evaluate the effectiveness of our proposed system, a series of experiments are conducted using two publicly available manufacturing datasets. Our evaluation focused on the system's ability to interpret user queries, generate correct and insightful answers, and maintain context over multi-turn conversations. We benchmarked these capabilities against direct prompting of LLMs, measuring accuracy, response time, and token usage.

### *3.1. Datasets and Question Design*

Two publicly available datasets are selected to represent various manufacturing scenarios. The first dataset, the Predictive Maintenance Dataset (AI4I2020) [29], consists of 10,000 samples with 14 features related to machine performance and failure. It includes sensor readings such as air temperature, process temperature, torque, rotational speed, and tool wear, along with product quality labels categorized as Low, Medium, or High. The dataset also features five types of failure modes: tool wear failure, heat dissipation failure, power failure, overstrain failure, and random failure. A binary target label indicates whether a machine failure occurred, although the specific failure type is not disclosed.

The second dataset, the Intelligent Manufacturing Dataset (Man6GData) [30], is a synthetic dataset designed to model smart manufacturing environments. It comprises real-time sensor readings, network metrics, and production performance data from intelligent manufacturing systems. It includes machine identifiers, operational parameters (e.g., temperature, vibration, power usage), 6G network indicators (latency, packet loss), and efficiency metrics such as defect rates and predictive maintenance scores. A target variable, Efficiency Status, classifies overall efficiency into High, Medium, or Low categories.

To evaluate reasoning and analytical capabilities of the proposed system, a benchmark set of natural language questions is created inspired by TableBench's [27] taxonomy of table-based queries. TableBench is a challenging benchmark for assessing Table Question Answering (TableQA) systems, with a strong focus on the reasoning complexity of questions. It spans 18 question types across four main areas including fact checking, numerical reasoning, data analysis, and visualization, and includes 886 well-curated test cases. The benchmark is designed to test and advance the capabilities of large language models in complex TableQA tasks. Specifically, 100 questions are created for each dataset. The choice of 100 questions per dataset balances statistical reliability and efficiency, and is larger than what prior works typically used (e.g., 9 questions in ChantCNC [19], 55 in ChatWithMes[31]). This scale also exceeds practical recommendations for complex agents [32], while remaining feasible for test-time evaluation compared to ~1000 samples often used for fine-tuning. Each question was paired with a corresponding reference answer and were created based on patterns derived from TableBench, then refined for domain relevance. Each question was paired with a reference answer, validating through Python-based data exploration and analysis scripts to ensure accuracy and grounding in the actual dataset. Some sample questions and answers are presented in Table 1, along with their corresponding categories. These categories highlight diverse analytical approaches and demonstrate the comprehensiveness

Table 1 Samples of created questions and answers, along with their corresponding categories, inspired by the TableBench [21] benchmark on AI4I2020.

| | | |
|---|---|---|
| Fact Checking | Match Based | **Question:** What is the maximum rotational speed [rpm] in the dataset? |
| | | **Answer:** 2886 |
| | Multi-hop Fact Checking | **Question:** What is the maximum torque [Nm] for products with a tool wear [min] less than 10? |
| | | **Answer:** 76.6 |
| Numerical Reasoning | Arithmetic Calculation | **Question**: What is the ratio of machine failures to total entries in the dataset? |
| | | **Answer**: 3.39 |
| | Multi-hop Numerical Reasoning | **Question**: How many entries have an air temperature [K] greater than 299 and a rotational speed [rpm] less than 1400? |
| | | **Answer**: 1159 |
| Data Analysis | Statistical Analysis | **Question**: What is the variance of torque for power failures? |
| | | **Answer**: 717.6319081746921 |
| | Correlation Analysis | **Question**: Is there a significant correlation between process temperature and machine failure? |
| | | **Answer**: The correlation coefficient between process temperature and machine failure is 0.04, indicating a weak positive relationship. |
| Visualization | Chart Generation | **Question**: Can you plot a chart to show the proportion of machine failures caused by different failure types? |
| | | **Answer**: TWF: 46, HDF: 115, PWF: 95, OSF: 98, RNF: 19 |

of the benchmark. The distribution of query types reflects their relative importance in manufacturing analysis, with numerical reasoning and fact-checking forming the majority, complemented by fewer data analysis and visualization queries.

### *3.2. Experimental Setup*

The system is evaluated using a set of LLMs including ‘GPT-3.5-Turbo’, ‘Gemini-2.0-Flash’ and ‘Llama-3.3-70B-Instruct-Turbo’, accessed either locally or via API depending on model. The goal was to understand how different LLMs perform when integrated into our agentic LangGraph-based workflow designed for manufacturing data analysis. To evaluate the system, we used the full set of benchmark questions created for each dataset, meaning 100 questions per dataset.

Each system response was assessed across three dimensions: correctness or accuracy, completion time for answering all questions, and cost, measured in terms of the number of tokens consumed. Two main approaches are compared: our proposed LangGraph-based agentic system and a direct LLM prompting baseline, in which the same queries were submitted directly to the language model without any agentic workflow or tool orchestration.

In the baseline method, the LLM was provided with the required information about the dataset along with the question, and it was requested to generate the corresponding SQL query. After receiving the generated query from the LLM, the query was executed separately, and the final answer was evaluated. In this experiment, the focus is more on comparing the data extraction capability rather than the visualization capabilities of the system. Moreover, the baseline approach has limitations in visualization; the required diagrams must be created separately and still require expertise in data analysis.

### *3.3. Results*

The results of our evaluation are summerized in Table 2 and Table 3, corresponding to the AI4I2020 and Man6GData benchmark questions, respectively. These tables compare the performance of the two approaches, our LangGraph-based agentic system and the direct LLM prompting baseline, across three key criteria: accuracy, execution time, and cost (measured in token usage).

Across both datasets, the agentic system consistently outperformed direct prompting in terms of accuracy, achieving an average improvement of 20 percentage points, depending on the LLM used. This gain is attributable to the system’s ability to understand complex queries, apply appropriate tools, and maintain conversation context across multiple turns.

Accuracy was measured through an automated comparison between the generated answers and reference values; for this purpose, the LLM-as-judge approach was used. By prompting ‘GPT-4o-mini’, two evaluation criteria were defined to compare each method's answer with the reference answer: a strict evaluation and a more flexible evaluation. In the prompt, the LLM-as-judge was instructed to evaluate the generated answers according to both criteria, assigning a score of 0 or 1 if the generated answer was exactly and strictly equal to the reference, and 0 or 1 if, from a flexible view, the generated and reference answers could reasonably be considered equivalent. Explanations were also requested for both scoring criteria. The scores and explanations were then manually reviewed to ensure that the LLM-as-judge was providing reliable evaluations, only a few adjustments were needed, mostly in borderline cases where an answer differed from the ground truth but was still reasonably correct. Finally, the reported accuracy in the Table 2 and 3 are based on the flexible criteria. The experiments demonstrate that agentic reasoning improves the system’s ability to handle complex or ambiguous queries, highlighting

Table 2 Result of evaluation and comparison of proposed model and baseline method on AI4I2020 benchmark questions. Accuracy is calculated as the number of correct results divided by the total number of queries, multiplied by 100. The reported time and token numbers are averaged across all benchmark samples.

| AI4I2020 | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | GPT-3.5-Turbo | | | Gemini-2.0-Flash | | | Llama-3.3-70B-Instruct-Turbo | | |
| | Accuracy | Time (s) | #Tokens | Accuracy | Time (s) | #Tokens | Accuracy | Time (s) | #Tokens |
| Baseline | 64 | 0.75 | 0.3K | 69 | 0.64 | 0.4K | 72 | 1.46 | 0.4K |
| Our Model | 90 | 2.12 | 2.3 K | 89 | 1.6 | 2.2 K | 90 | 3.45 | 2.4K |

Table 3 Result of evaluation and comparison of proposed model and baseline method on Man6GData benchmark questions. Accuracy is calculated as the number of correct results divided by the total number of queries, multiplied by 100. The reported time and token numbers are averaged across all benchmark samples.

| Man6GData | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | GPT-3.5-turbo | | | Gemini-2.0-Flash | | | Llama-3.3-70B-Instruct-Turbo | | |
| | Accuracy | Time (s) | #Tokens | Accuracy | Time (s) | #Tokens | Accuracy | Time (s) | #Tokens |
| Baseline | 59 | 0.74 | 0.4K | 69 | 0.62 | 0.4K | 70 | 1.51 | 0.4K |
| Our Model | 83 | 2.49 | 2.1K | 86 | 1.51 | 1.9K | 87 | 4.74 | 2.3K |

the potential of LLMs, especially when compared to direct prompting. By orchestrating tool usage, maintaining context, and leveraging modular workflows, the LangGraph-based agentic system delivers more accurate and interpretable data analysis in manufacturing settings. The use of a structured benchmark enables systematic, repeatable evaluation across datasets and LLM types.

## 4. Discussions and Conclusions

This paper introduced a novel lightweight agentic AI system that enables non-expert users to perform complex manufacturing data analysis through natural language interaction. Built on the LangGraph framework, the system integrates large language models with modular tool orchestration to support accurate, interpretable, and context-aware data workflows. Using two manufacturing datasets, it is demonstrated that our system can effectively interpret complex queries, enabling non-expert users to extract meaningful insights from manufacturing data through conversational interaction. Our experimental results show that agentic reasoning significantly improves performance compared to direct LLM prompting, supporting real-world use. Across models, GPT-3.5, Gemini, and LLaMA-3 show consistent gains (Tables 2 and 3), with LLaMA-3 yielding slightly higher accuracy but longer runtime. We attribute this to LLaMA-3's deeper reasoning capacity, which comes at the cost of efficiency, whereas GPT-3.5 and Gemini maintain faster responses with marginally lower accuracy. Notably, agentic reasoning reduced performance variance across repeated trials, suggesting it stabilizes outcomes independent of model choice. These results imply that model selection in practice will hinge on whether accuracy or efficiency is prioritized.

The proposed approach illustrates how agentic reasoning can bridge the gap between technical data infrastructure and everyday decision-making, marking a step forward in the application of LLMs for industrial analytics. In practice, such systems can reduce analysis time, democratize access to complex datasets, and augment shop-floor decision-making. Potential users include quality engineers, maintenance technicians, and production planners. As manufacturing increasingly depends on real-time, data-rich environments, systems like ours offer a scalable and user-friendly pathway to integrate AI into daily operations. Continued refinement and validation in real-world settings will further enhance its practical value and impact.

However, several limitations were observed. The system occasionally struggled with ambiguous or underspecified queries, particularly when clarification was needed but not explicitly requested. Additionally, execution inconsistencies in the query (or code) also impacted output and system reliability. Another limitation concerns scalability, the current system may struggle with very large datasets unless further optimization strategies and big data analysis techniques are integrated. In addition, our current evaluation is based on publicly available benchmark datasets (AI4I2020 and Man6GData) to ensure reproducibility, future work will extend validation to more heterogeneous, real-world manufacturing data. In contrast, the baseline method often failed due to (1) SQL queries that triggered execution errors and (2) incorrect results stemming from an incomplete understanding of complex questions. Our system addresses these issues through orchestration-layer control and conversational feedback, enabling more robust handling of ambiguity and errors. While this study focused on single-query analysis, the benefits of our approach are even more evident in multi-turn conversational scenarios.

Also, regarding the higher completion time and number of tokens used in the proposed model compared to the baseline, this increase is primarily due to the agentic workflow and the multiple steps involved in the process. These additional steps naturally lead to longer processing times and a greater number of token exchanges between the modules of the agentic system. However, in real-world applications, this trade-off is acceptable given the significantly greater potential and capabilities of the agentic workflow. The improved flexibility, robustness, and ability to handle complex tasks justify the higher time and token cost.

It is worth noting that commercial platforms such as ChatGPT, Gemini, and similar tools also offer data analysis capabilities through file uploads. However, these solutions often face limitations in cost efficiency, data confidentiality, and scalability. Specifically, analyzing large datasets can become prohibitively expensive, pose risks to sensitive data, or be infeasible due to platform limitations on file size. In contrast, our lightweighted agentic approach offers a more cost-effective and privacy-preserving alternative. It is designed to operate securely within local or controlled environments and is compatible with rather large-scale datasets, making it a practical and scalable solution for industrial applications.

Future works point towards user studies with domain experts to assess usability and relevance in real-world settings. Moreover, expanding the system to handle real-time data from streaming sources is a critical next step toward fully enabling AI-assisted analytics in intelligent manufacturing contexts.

For practical deployment, the system can be further developed to autonomously identify and apply appropriate data analysis tasks on a given dataset, generate comprehensive analytical reports, and highlight key findings. Such capabilities would significantly streamline the data analysis process, which is an essential component of daily operations in the manufacturing industry, and finally enhance decision-making efficiency. It is also important to note that evaluating such systems remains a critical and inherently challenging task.

## Acknowledgements

This research was supported by the Independent Research Fund Denmark or Danmarks Frie Forskningsfond (DFF) [RECODE, case number 3163-00006B].

## References

[1] Yao S, Zhao J, Yu D, Du N, Shafran I, Narasimhan K, et al. ReAct: Synergizing Reasoning and Acting in Language Models. International Conference on Learning Representations (ICLR) 2023.

[2] Shen Z. LLM With Tools: A Survey. ArXiv Preprint ArXiv:240918807 2024.

[3] Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N, et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. Adv Neural Inf Process Syst 2020;33:9459–74.

[4] Shinn N, Cassano F, Gopinath A, Narasimhan K, Yao S. Reflexion: language agents with verbal reinforcement learning. Adv Neural Inf Process Syst 2023;36:8634–52.

[5] Jimenez CE, Yang J, Wettig A, Yao S, Pei K, Press O, et al. SWE-bench: Can Language Models Resolve Real-World GitHub Issues? 12th International Conference on Learning Representations, ICLR 2024.

[6] Jha S, Arora R, Watanabe Y, Yanagawa T, Chen Y, Clark J, et al. ITBench: Evaluating AI Agents across Diverse Real-World IT Automation Tasks. International Conference on Machine Learning 2025.

[7] Levy I, Wiesel B, Marreed S, Oved A, Yaeli A, Shlomov S. ST-WebAgentBench: A Benchmark for Evaluating Safety and Trustworthiness in Web Agents. International Conference on Machine Learning (ICML) 2025 Workshop on Computer Use Agents 2024.

[8] Pan Y, Kong D, Zhou S, Cui C, Leng Y, Jiang B, et al. WebCanvas: Benchmarking Web Agents in Online Environments. Agentic Markets Workshop at ICML 2024.

[9] Laurent JM, Janizek JD, Ruzo M, Hinks MM, Hammerling MJ, Narayanan S, et al. LAB-Bench: Measuring Capabilities of Language Models for Biology Research. ArXiv Preprint ArXiv:240710362 2024.

[10] Nathani D, Madaan L, Roberts N, Bashlykov N, Menon A, Moens V, et al. MLGym: A New Framework and Benchmark for Advancing AI Research Agents. ArXiv Preprint ArXiv:250214499 2025;5.

[11] Castillo-Bolado D, Davidson J, Gray F, Goodai R. Beyond Prompts: Dynamic Conversational Benchmarking of Large Language Models. Advances in Neural Information Processing Systems, 37 2024:42528–65.

[12] Plurai EL, Plurai IK. IntellAgent: A Multi-Agent Framework for Evaluating Conversational AI Systems. ArXiv Preprint ArXiv:250111067 2025.

[13] Sun M, Han R, Jiang B, Qi H, Sun D, Yuan Y, et al. A Survey on Large Language Model-based Agents for Statistics and Data Science. Causal and Object-Centric Representations for Robotics Workshop at CVPR 2024.

[14] Zhang L, Zhang Y, Ren K, Li D, Yang Y. MLCopilot: Unleashing the Power of Large Language Models in Solving Machine Learning Tasks. EACL 2024 - 18th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference 2023;1:2931–59.

[15] Sun M, Han R, Jiang B, Qi H, Sun D, Yuan Y, et al. LAMBDA: A Large Model Based Data Agent. Journal of the American Statistical Association Just-Accepted 2025:1-20.

[16] Zhang W, Shen Y, Tan Z, Hou G, Lu W, Zhuang Y. Data-Copilot: Bridging Billions of Data and Humans with Autonomous Workflow. The International Conference on Learning Representations (ICLR) Workshop on Large Language Model (LLM) Agents 2024.

[17] Luo D, Feng C, Nong Y, Shen Y. AutoM3L: An Automated Multimodal Machine Learning Framework with Large Language Models. MM 2024 - Proceedings of the 32nd ACM International Conference on Multimedia 2024:8586–94. https://doi.org/10.1145/3664647.3680665.

[18] Gautam A, Aryal MR, Deshpande S, Padalkar S, Nikolaenko M, Tang M, et al. IIoT-enabled digital twin for legacy and smart factory machines with LLM integration. J Manuf Syst 2025;80:511–23. https://doi.org/10.1016/J.JMSY.2025.03.022.

[19] Jeon J, Sim Y, Lee H, Han C, Yun D, Kim E, et al. ChatCNC: Conversational machine monitoring via large language model and real-time data retrieval augmented generation. J Manuf Syst 2025;79:504–14. https://doi.org/10.1016/J.JMSY.2025.01.018.

[20] Gkournelos C, Konstantinou C, Makris S. An LLM-based approach for enabling seamless Human-Robot collaboration in assembly. CIRP Annals 2024;73:9–12. https://doi.org/10.1016/J.CIRP.2024.04.002.

[21] Ferrag MA, Tihanyi N, Debbah M. From LLM Reasoning to Autonomous AI Agents: A Comprehensive Review. ArXiv Preprint ArXiv:250419678, 2025.

[22] Malhan R, Gupta SK. The Role of Deep Learning in Manufacturing Applications: Challenges and Opportunities. J Comput Inf Sci Eng 2023;23. https://doi.org/10.1115/1.4062939.

[23] Wan Y, Chen Z, Liu Y, Chen C, Packianather M. Empowering LLMs by hybrid retrieval-augmented generation for domain-centric Q&A in smart manufacturing. Advanced Engineering Informatics 2025;65:103212. https://doi.org/10.1016/J.AEI.2025.103212.

[24] Douimia S, Bekrar A, Ait El Cadi A, El Hillali Y, Fillon D. Machine learning and deep learning applications in the automotive manufacturing industry: A systematic literature review and industry insights. Robot Comput Integr Manuf 2025;96:103034. https://doi.org/10.1016/J.RCIM.2025.103034.

[25] Bezerra A, Greati V, Campos V, Silva I, Guedes LA, Leitão G, et al. Enabling Interactive Visualizations in Industrial Big Data. IFAC-PapersOnLine 2020;53:11162–7. https://doi.org/10.1016/J.IFACOL.2020.12.292.

[26] Wei W, Yuan J, Liu A. Manufacturing data-driven process adaptive design method. Procedia CIRP 2020;91:728–34. https://doi.org/10.1016/J.PROCIR.2020.02.230.

[27] Wu X, Yang J, Chai L, Zhang G, Liu J, Du X, et al. TableBench: A Comprehensive and Complex Benchmark for Table Question Answering. Proceedings of the AAAI Conference on Artificial Intelligence 2024;39:25497–506. https://doi.org/10.1609/aaai.v39i24.34739.

[28] Satyadhar Joshi. Review of autonomous systems and collaborative AI agent frameworks. International Journal of Science and Research Archive 2025;14:961–72.

[29] Matzka S. Explainable Artificial Intelligence for Predictive Maintenance Applications. Proceedings - 2020 3rd International Conference on Artificial Intelligence for Industries, AI4I 2020:69–74.

[30] Intelligent Manufacturing Dataset n.d. https://www.kaggle.com/datasets/ziya07/intelligent-manufacturing-dataset (accessed July 9, 2025).

[31] Yuan Z, Li M, Liu C, Han F, Huang H, Dai HN. Chat with MES: LLM-driven user interface for manipulating garment manufacturing system through natural language. J Manuf Syst 2025;80:1093–107. https://doi.org/10.1016/J.JMSY.2025.02.008.

[32] Agents - Evaluations n.d. https://docs.uipath.com/agents/automation-cloud/latest/user-guide/agent-evaluations (accessed September 4, 2025).