



Foundation language models through the lens of manufacturing

Sareh Aghaei & Fazel Ansari

To cite this article: Sareh Aghaei & Fazel Ansari (2026) Foundation language models through the lens of manufacturing, *Production & Manufacturing Research*, 14:1, 2632468, DOI: [10.1080/21693277.2026.2632468](https://doi.org/10.1080/21693277.2026.2632468)

To link to this article: <https://doi.org/10.1080/21693277.2026.2632468>



© 2026 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.



Published online: 20 Feb 2026.



Submit your article to this journal [↗](#)



Article views: 412




View related articles [↗](#)



View Crossmark data [↗](#)

Foundation language models through the lens of manufacturing

Sareh Aghaei and Fazel Ansari 

Chair of Production and Maintenance Management, Faculty of Mechanical and Industrial Engineering, TU Wien, Vienna, Austria

ABSTRACT

Although recent studies have focused on foundation language models' architectures, scaling properties, and applications in fields such as healthcare and business, no detailed investigation has addressed their role in manufacturing. This paper fills this gap by examining foundation language models, with a particular focus on large language models (LLMs) as their most prominent instantiation, through the operational manufacturing lens, emphasizing their capabilities and practical applications. In the first part, the core capabilities of LLMs are categorized and analyzed. These capabilities include text understanding and generation, reasoning, multi-modality, interactivity, generalization, and continual learning. The second part examines how these capabilities translate into practical applications across the operational phases of manufacturing. The areas include planning, production, material handling, engineering, quality, maintenance, and warehousing. By aligning LLM functionalities with operational manufacturing phases, the paper shows LLMs' potential to augment decision-making, enhance efficiency, and increase adaptability in the context of Industry 4.0.

ARTICLE HISTORY

Received 10 November 2025
Accepted 11 February 2026



KEYWORDS

Large language models;
manufacturing systems;
operational manufacturing
phases; industry 4.0;
decision-making
augmentation

1. Introduction

The advent of foundation language models has brought about a paradigm shift in natural language processing (NLP), demonstrating remarkable abilities in contextual understanding and content generation (Naveed et al., 2025). Language models are computational systems with the ability to predict the probability of word sequences or generate new text based on textual input data, such as N -gram models serving as early examples (Chang et al., 2024). Large language models (LLMs) are a subset of neural models, including huge parameter sizes and exceptional learning capabilities (Chang et al., 2024; Naveed et al., 2025; Vaswani et al., 2017). LLMs represent a prominent instantiation of foundation language models, characterised primarily by their scale and versatility. The most current LLMs use the Transformer architecture (Vaswani et al., 2017), whose main novelty is its attention mechanism. The attention mechanism facilitates the ability of language models to handle long-range dependencies in textual data, identifying the relevance of different parts of the input when generating predictions (Vaswani et al., 2017). This allows the models to understand the dependencies between words in a sentence, independent of their position.

Industry 4.0 integrates machines and industrial processes into cyber-physical production systems through advanced information and communication technologies (Vogel-Heuser & Hess, 2016). Industry 4.0 uses these advanced technologies to offer digital solutions and smart objects, involving humans, data, services, machines, and things (Zhang et al., 2024a). It provides continuous communication among humans, machines, and manufacturing systems during the production process (Zhang et al., 2024a). Industry 4.0 shifts centralised production to a decentralised model through shared facilities in integrated global industrial systems (Zhang et al., 2024a). Industry 4.0 was originally conceived as the integration of cyber-physical systems, automation, and digital connectivity enabled by technologies such as the Internet of Things (IoT), cloud computing, and Big Data analytics. Although artificial intelligence (AI) does not

CONTACT Sareh Aghaei  sareh.ghaei@tuwien.ac.at  TU Wien, Institute of Management Science Theresianumgasse 27, 1040, Vienna, Austria

© 2026 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.
This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

explicitly appear in its initial formulation, it plays a supporting rather than central role, primarily as an enabler within the Big Data and analytics pillar.

As a key component of AI technologies, LLMs are increasingly revealing their significant potential in Industry 4.0. A cascade of LLM applications has emerged across various sectors of Product Lifecycle Management (PLM), spanning from initial concept design to end-of-life management and disposal, with particular focus on the operational phase of manufacturing.

While most surveys focus primarily on the fundamentals of LLMs, there remains a need to bridge the gap between their algorithmic development and their practical deployment in industrial contexts. For instance, (Matarazzo & Torlone, 2025) explores foundational components, scaling, and data growth impacts but lacks emphasis on industrial applications. Similarly, (Annepaka & Pakray, 2025) reviews LLM evolution, transformer history, and diverse domains such as business and healthcare. The survey in (Kumar, 2024) covers language modelling, architectures, and applications in fields such as biomedicine, but also omits industrial use, underscoring a gap in understanding the role of LLMs in production-oriented environments. Likewise, (Urlana et al., 2024) focuses on datasets, evaluation, and deployment, without fully addressing real-world industrial applications.

Therefore, there is a need for a dedicated investigation that systematically examines how LLMs are being, or can be, applied within manufacturing environments, with a particular emphasis on their capabilities in addressing domain-agnostic challenges across the operational phases of manufacturing. This paper focuses on the operational phase of product lifecycle management (PLM), encompassing activities from planning to warehousing. To structure the analysis, the paper adopts the operational manufacturing categorisation defined in (Baudin & Netland, 2022), distinguishing seven core functions: planning, production, material handling, engineering, quality, maintenance, and warehouse. According to this analysis, planning develops production plans based on forecasts and incoming sales orders. Production manufactures parts or products according to plans, converting raw materials into finished goods. Materials handling ensures that production always has the necessary materials, managing dock-to-dock logistics within a plant. Engineering is responsible for designing, creating, and updating the layout of the factory, machinery, production lines, and information systems. Quality ensures that products meet customer expectations through effective quality control methods. Maintenance aims to maintain machinery and tools in working condition and to handle repairs that exceed the capabilities of operators. Warehousing stores and ships finished products, ensuring timely delivery.

2. Research methodology

The Structured Literature Review (SLR) methodology (Keele, 2007) is adopted to systematically explore LLMs in the context of manufacturing applications. Upon a defined topic, the SLR identifies, evaluates, and synthesises all relevant research. To ensure transparency and rigour in the research selection process, the following subsections outline the research methodology, including the Research Questions (RQs) and the procedure for paper selection.

2.1. Question formulation

The paper pursues two main objectives. The first aims to identify and analyse the core capabilities of LLMs, particularly their applicability in addressing challenges within practical domains. The second seeks to trace the emerging thematic areas surrounding the adoption of LLMs in manufacturing, examining key application areas, trends, and advancements that define their role in manufacturing processes. Therefore, the primary research question of this paper is *What fundamental properties and capabilities enable LLMs to contribute to complex problem solving, and how do their applications demonstrate their ability to address industrial domain-agnostic challenges in the operational phase of manufacturing, from planning to warehousing*. Based on this, two sub-research questions are derived as follows:

- (1) What technical characteristics and capabilities of LLMs contribute to their effectiveness in the operational phase of manufacturing?

- (2) How are LLMs currently being applied at different stages of operational manufacturing, including planning, production, material handling, engineering, quality, maintenance, and warehouse?

Subsequently, [Section 3](#) examines the capabilities of LLMs to address the first sub-research question, while [Section 4](#) explores their applications across manufacturing domains in response to the second.

2.2. Paper selection procedure

The paper selection procedure follows explicit inclusion and exclusion criteria to ensure methodological rigour and relevance. Inclusion criteria require studies to be peer-reviewed journal articles or conference papers written in English that explicitly applied or evaluated LLMs in operational manufacturing and industrial contexts. Exclusion criteria include studies focused solely on generic NLP tasks addressed non-industrial domains, or works lacking sufficient methodological detail.

To identify relevant English-language literature addressing the sub-research questions, three major academic databases, including Scopus, ScienceDirect, and ACM Digital Library, are selected for their comprehensive coverage of interdisciplinary research in computer science, engineering, and industrial applications. The search focuses on the period from 2017 to 2025, as the attention mechanism, a fundamental building block of almost all LLMs, was introduced in 2017 (Vaswani et al., 2017). A search syntax is formulated, along with inclusion and exclusion criteria. The search syntax is built using the following keywords: ‘language models’; ‘manufacturing’; ‘industrial’; ‘reasoning’; ‘code generation’; ‘continual learning’; and ‘capabilities’. The search syntax has to include the term ‘language model’ and at least one other keyword. A total of 900 papers are identified based on keyword searches in the databases. Specifically, 733 papers are retrieved from Scopus, 36 papers from the ACM Digital Library, and 131 papers from ScienceDirect. The duplicate studies are then identified and removed using the Parsif tool, resulting in 771 unique articles.

The screening process was conducted in two phases. In the first phase, the titles, abstracts, and keywords of 771 English, non-duplicate papers were reviewed. Based on the predefined inclusion and exclusion criteria, 615 papers were excluded because they were not peer-reviewed journal articles or conference papers that explicitly applied or evaluated large language models in operational manufacturing contexts. Studies were also excluded if they focused solely on generic natural language processing tasks, addressed non-industrial domains, or fell outside the scope of operational manufacturing. This screening resulted in 149 studies retained for further consideration. Backward citation chaining was then applied to these 149 studies to identify additional relevant publications not captured in the initial search, adding 53 studies and increasing the candidate set to 202 studies. In the second phase, the full texts of these 202 studies were examined, and the same inclusion and exclusion criteria were applied in greater depth. At this stage, 118 studies were excluded because they lacked sufficient methodological detail or did not provide a substantive application or evaluation of LLMs in manufacturing contexts. This resulted in a final corpus of 84 studies included in the review.

3. LLM capabilities

This section describes the capabilities of LLMs across key functional areas relevant to both research and industrial applications, including text understanding and generation, reasoning, multi-modality, interactivity, and generalisation. These LLM capabilities were derived through a systematic literature review rather than being arbitrarily defined. The capability-related terms were initially used as search keywords, and the resulting studies were subsequently qualitatively analysed to validate and refine the capability set based on recurring functional behaviours reported in the literature. Each capability reflects the evolving potential of LLMs to perform complex tasks, and augment intelligent decision-making in diverse contexts, including manufacturing.

3.1. Understanding

The ability of understanding in LLMs is broadly divided into two domains: text understanding and code understanding.

Text understanding. The capability to learn successfully from textual data is essential to reduce the dependence on supervised learning or hand-crafted rules in NLP (Radford & Narasimhan, 2018). The text understanding capability of LLMs arises from self-supervised learning, enabling them to acquire linguistic knowledge directly from textual data. This approach allows them to learn rich syntactic and semantic representations without explicit annotations, due to their transformer-based architecture (Chang & Bergen, 2024; Vaswani et al., 2017). In particular, the encoder component of the transformer plays a central role in text understanding by focusing on processing and comprehending textual sequences. The key sub-components of the encoder include positional encoding, self-attention mechanism, and multi-head attention. Positional encoding provides order (or positional) information to token representations, while self-attention mechanism enables the model to capture the contextual dependencies of tokens within a sequence. Multi-head attention supports LLMs to learn multiple aspects of word relationships simultaneously, enhancing their ability to understand complex linguistic structures and patterns (Chang & Bergen, 2024; Vaswani et al., 2017).

Code understanding. Code understanding involves learning code syntax and semantics of a programming language. Semantics refers to how a code behaves during execution, including how inputs are processed, computed, and outputted in order to ensure that it functions correctly (Ding et al., 2024).

From an architectural perspective, the code understanding capability often resides in the encoder component, where simplicity, universality, and scalability contribute to enhancing this capability (Nijkamp et al., 2023). CodeBERT, Codex, StarCoder, and CodeGen are examples of LLMs specifically trained for coding tasks. Models that rely on encoder-only architectures, such as CodeBERT, are well-suited to tasks like type prediction, code retrieval, and clone detection in the context of code comprehension (Jiang et al., 2024).

3.2. Generation

The generative capability of LLMs is commonly divided into two domains: text generation and code generation.

Text generation. A text can be defined as a sequence of tokens, each drawn from a vocabulary, and text generation aims to produce fluent, human-readable sequences with prior context (Li et al., 2024a). LLMs for text generation are categorised into two main types: decoder-only LLMs and encoder-decoder LLMs based on the architecture. Decoder-only LLMs (e.g. GPT-series models) are specifically designed for language modelling, where the objective is to predict the next token based on previous tokens. In contrast, encoder-decoder LLMs (e.g. T5 and BART) follow the standard transformer architecture, consisting of stacks of encoder and decoder layers (Li et al., 2024a). In addition to pre-training and architectural design, optimisation techniques such as fine-tuning and prompt-tuning enhance the text-generation capabilities of LLMs (Li et al., 2024a).

Code generation. Code generation is the process of creating a code snippet based on a user's intent, expressed through a natural language requirement. LLMs offer significant potential for code generation, reducing manual coding efforts, ensuring consistency, and minimising the risk of human error during development (Liu et al., 2024a; Mu et al., 2024). A key training ground for LLMs to develop coding proficiency is large-scale unlabelled code datasets such as CodeSearchNet, Google BigQuery, Stack Overflow, and GitHub (Jiang et al., 2024). Decoder-only models, such as StarCoder and the GPT series, are primarily designed for autoregressive code generation, excelling in tasks such as code synthesis, code translation, and code summarisation (Ahmad et al., 2023). These models are trained with the objective of the next-token prediction, generating code sequentially from left to right in a unidirectional manner (Mu et al., 2024). In contrast, encoder-decoder models, such as CodeT5, PLBART, and AlphaCode, enable handling both code understanding and code generation tasks; however, they do not necessarily outperform encoder-only or decoder-only models in either category (Jiang et al., 2024; Mu et al., 2024).

3.3. Reasoning

As a term, reasoning refers to using stated premises to arrive at conclusions and generalising those patterns to new contexts (Chen et al., 2024a). LLMs' reasoning capabilities arise from the architecture, pre-training, fine-tuning, in-context learning, prompt engineering, and integration with external knowledge bases.

Pre-training is an important phase for developing reasoning abilities, particularly when training datasets contain rich math, and code examples (Xu et al., 2025). Following pre-training, supervised fine-tuning enables models to perform better in zero-shot and more complex reasoning tasks. An advanced approach to optimising LLM reasoning during fine-tuning involves reinforcement learning-based training at the final alignment stage of LLM development (Xu et al., 2025). Moreover, LLMs provide reasoning abilities through in-context learning (Xu et al., 2025). In-context learning refers to learning from a few examples in a context similar to learning from analogies in human decision-making (Dong et al., 2024). In cases where zero-shot reasoning remains unreliable, prompt engineering can enhance reasoning capabilities (Ho et al., 2023; Xu et al., 2025; Xu et al., 2025). Additionally, incorporating external knowledge (e.g. a knowledge graph) increases the ability of LLMs to perform logical inferences (Chen et al., 2024a).

Recent foundational works have advanced the reasoning capabilities of LLMs (Guo et al., 2025; Huang & Chang, 2022; Rastogi et al., 2025). For instance, DeepSeek's initial approach (Guo et al., 2025) introduced Group Relative Policy Optimisation (GRPO), a light reinforcement learning algorithm designed to enhance reasoning ability autonomously, without extensive supervised fine-tuning. Furthermore, DeepSeek employs a Mixture of Experts (MoE) architecture (Liu et al., 2024b) composed of multiple experts or sub-networks. The experts specialise in various tasks or features. A dynamic routing mechanism directs input representations to the most relevant experts, improving computational efficiency and task specialisation.

3.4. Multi-modality

With multi-modal LLMs, text is paired with other modalities (such as images, audio recordings), enabling cross-modal understanding and generation (Tamkin et al., 2021). As a result, models are able to obtain capabilities faster, since the interaction between different data modalities provides a stronger learning signal than each modality alone (Tamkin et al., 2021). Multi-modal LLMs are divided into early-fusion (such as VisualGPT and DeepMind's Perceiver) and alignment architectures (such as GPT-4V and Sora) (Chen et al., 2024b). Early-fusion architectures tokenize visual inputs through a visual tokenizer, much like text tokenization. A multi-modal autoregressive language model then processes both textual and visual tokens together to generate the desired output (Chen et al., 2024b). Alignment architectures, on the other hand, which are commonly used in practice, align the vision modality (or other modalities) with a pre-trained LLM (Chen et al., 2024b). These models create embeddings for visual inputs through vision-language pre-training. The embeddings are later mapped into the LLM's representation space using components such as projectors or Q-Formers (Chen et al., 2024b).

3.5. Interactivity

The term interactivity generally implies the involvement of humans. Although humans are the primary agents interacting with language models, recent research has expanded the scope to include additional objects such as knowledge bases, models, tools, and environments (Wang et al., 2023). Therefore, interactive NLP treats LLMs as agents that observe, act, and receive feedback in a continuous loop with these objects (Wang et al., 2023).

Human-in-the-loop NLP prioritises continuous interaction between humans and LLMs to meet user needs while maintaining human values (Wang et al., 2023). Communicating with human prompts, learning from human feedback, regulating via human configuration, and learning from human simulation are strategies to enhance human-language model alignment (Wang et al., 2023).

In knowledge base-in-the-loop NLP, external knowledge is used during training or inference to improve LLM accuracy and context-awareness (Wang et al., 2023). Interactivity with models and tools means to decompose a complex problem into modular sub-tasks. Then, each sub-task is assigned to a specialised language model agent, expert model, or external tool. Thus, adaptive task execution supports efficiency and accuracy in problem-solving (Wang et al., 2023).

Environment-in-the-loop focuses on language grounding by integrating the real or virtual environment into an interactive loop with LLMs. In this interaction, the environment provides the model with low-level observations, rewards, and state transitions. The model solves environment tasks, such as reasoning, planning, and decision-making (Li et al., 2022; Wang et al., 2023).

3.6. Continual learning

Continual learning (or lifelong learning or incremental learning) refers to a model's ability to continuously acquire and integrate new knowledge over time without forgetting its previous knowledge (Yang et al., 2024). LLMs learn continuously through forward transfer, utilising knowledge from previous tasks to learn new tasks without catastrophic forgetting (i.e. the loss of previously acquired knowledge when learning new tasks) (Yang et al., 2024).

Replay-based methods, regularisation-based methods, architecture-based methods, and distillation-based methods are various techniques that enhance the continual learning capability of LLMs (Yang et al., 2024). Data from previous tasks is introduced through sampling or synthesis in replay-based methods (Bohao et al., 2024). Regularisation-based methods penalise significant updates to model parameters, thereby maintaining prior knowledge (Yang et al., 2024; Zheng et al., 2024). Architecture-based methods, such as prompt-tuning or prefix-tuning, focus on adapting the model's structure to integrate new tasks while reducing disruption to previously acquired knowledge (Zheng et al., 2024). Distillation-based methods use a distillation loss that encourages reducing performance deviations between teacher and student networks. The teacher network represents the model learned from the previous task, and the student network is trained on the current task guided by the teacher (Li et al., 2023).

3.7. Generalisation

Generalisation refers to a model's ability to generalise from past experiences to new, and unseen situations. The focus of generalisation is to enable a model to perform well on unseen samples from the same distribution, whereas the focus of continual learning is to adapt the model to new tasks or data distributions while maintaining previously learned knowledge. The generalisation capabilities of LLMs stem from their architecture, pre-training, and fine-tuning processes and can be discussed from three aspects: length, structural, and cross-task generalisation (Li et al., 2024b).

Length generalisation in LLMs refers to the model's ability to apply acquired skills beyond its training set to longer problem instances. This supports to address complex problems with extensive descriptions as the model is able to manage longer input sequences (Li et al., 2024b). Structural generalisation refers to the capability to process and interpret complex data structures, such as graphs and tables, even when the models are trained on text-only datasets (Li et al., 2024b). Task generalisation refers to the ability to handle a wide range of tasks, especially those not seen during training (Li et al., 2024b).

4. Applications of LLM in manufacturing

This section describes how LLMs can be applied during the operational phase of manufacturing to support, automate, and enhance tasks within these areas, contributing to increased efficiency, accuracy, and adaptability in Industry 4.0. Rather than aiming to provide an exhaustive inventory of all LLM-based manufacturing applications, this section presents a curated and representative set of studies selected to illustrate how the core LLM capabilities identified in Section 3 manifest across different operational phases of manufacturing. This capability-driven selection supports a comparative analysis of application maturity and empirical evidence across phases, enabling analytical insights beyond a comprehensive enumeration of published work.

Following the categorisation proposed in (Baudin & Netland, 2022), the operational phase is divided into planning, production, material handling, engineering, quality, maintenance, and warehousing. Each of these areas involves data-intensive, decision-driven activities that can benefit from the advanced capabilities of LLMs discussed in section 3.

4.1. Planning

LLMs enable multi-agent systems to collaborate by supporting effective communication, coordination, shared reasoning, and negotiation among agents. Instead of traditional multi-agent systems that rely on predefined rules and rigid workflows, the Autogen framework (Barbosa et al., 2025) introduces and

develops an LLM-based approach where each agent is responsible for a specific step in the production process (e.g. designer agent, material sourcing agent, quality inspector agent). The agents are powered by context from an external repository. This study presents the outcomes of how this framework can be applied in manufacturing to address various scenarios that involve collaboration and negotiation.

According to the taxonomy proposed in (Pallagani et al., 2024), LLMs can be integrated into key planning dimensions. The defined dimensions include ‘language translation,’ ‘plan generation,’ ‘model construction,’ ‘multi-agent planning,’ ‘interactive planning,’ ‘heuristic optimisation,’ ‘tool integration,’ and ‘brain-inspired planning’. Planning scenarios involve translation of natural language descriptions into machine-understandable languages like PDDL (Planning Domain Definition Language). LLMs, particularly causal LLMs, are employed for plan generation (Sermanet et al., 2023) using in-context learning techniques such as chain-of-symbol and tree-of-thoughts. In the model construction dimension, LLMs are used to construct or refine domain models for planning, such as to simplify task roadmaps to make reinforcement learning more efficient (Li et al., 2024c). In the context of multi-agent planning (Zhang et al., 2024b), iterative planning is the key to adapting to user feedback in real time. Depending on the scenario, external verifiers, reinforcement learning, self-refinement techniques, and expert input can refine LLM outputs. The heuristic optimisation dimension (Hazra et al., 2024) involves refining existing plans using LLMs or providing heuristic guidance to symbolic planners by giving them heuristic guidance. Furthermore, LLMs enhance functionality in complex scenarios by coordinating various planning tools (Xu et al., 2023). Finally, brain-inspired planning (Mondal et al., 2024) investigates how LLMs can mimic human-like planning abilities using neurological and cognitive inspiration.

4.2. Production

Human-robot collaboration (HRC) is the interaction between human operators and robots within a shared workspace, serving as an enabling approach to production (Giallanza et al., 2024). Collaborative robots integrate industrial automation and cognitive human capabilities to improve production efficiency (Giallanza et al., 2024). As a result of this interaction, mass production is transformed into mass customisation, while human needs and well-being are prioritised (Giallanza et al., 2024).

Considering scheduling as part of production, in a manufacturing environment where materials and tools are not always located at the workstation, human operators often face interruptions in production due to tool retrieval. In this scenario, LLMs help robots fetch tools more efficiently, thereby reducing disruptions. For instance, a multi-modal cobot navigation framework is presented in (Wang et al., 2024a) to control automated guided vehicles (AGVs) in manufacturing. A 3D map of the workspace where assembly tasks are performed is created. Although operators use natural language commands to navigate, retrieve, and return tools, the commands are processed by GPT-3.5-turbo and converted into executable Python code for spatial navigation. A simulation of this framework is performed using AI Habitat. Moreover, a point-cloud model is annotated to provide spatial coordinates for objects within the HRC space and to facilitate effective navigation path planning. Using LLMs to enable robots to communicate more effectively with humans is also demonstrated in (Konstantinou et al., 2024; Li et al., 2024d; Lim et al., 2024). In the study presented in (Tsushima et al., 2025), operators’ requests are formulated as executable procedures using GPT-4.0, aligned with the robot’s capabilities. Also, the production line and safety-related operational constraints within the factory are considered to structure the prompts.

In intelligent industrial production, fixed control procedures are mostly incapable of adapting to changing production conditions, leading to high energy consumption, low productivity, and high operating costs. To address these challenges, an LLM-based system is developed using Langchain, METAGPT, and the Devin agent in (Wang & Qin, 2024). The developed system contains three main modules, including the data acquisition module, decision-making module, and human-computer interaction module. The data acquisition module collects real-time sensor data on equipment status, energy consumption, and environmental parameters. For optimisation, the decision-making module processes this data and incorporates expert input. Using the human-computer interaction module, operations can be monitored and adjusted intuitively. Different aspects of industrial production can be optimised with LLMs, as demonstrated in this study.

4.3. Materials handling

Analysis and evaluation of material handling are essential for decision-makers to optimise manufacturing and distribution operations by improving material flow management (Leung & Lau, 2020). The material handling processes often involve complex interactions which can be replicated, analysed, and predicted through simulation modelling using LLMs (Jackson et al., 2024). Jackson et al. (2024) uses GPT-3 Codex to generate simulation models for logistics systems by converting verbal descriptions into Python simulations. This streamlines the modelling process and reduces domain experts' workload. Additionally, it presents a collaborative framework between human experts and AI systems that emphasises clear role division, transparency, authority balance, and mutual learning—where humans provide input and verify results, and AI generates executable simulation code. Together, these elements enhance trust, accountability, and overall simulation modelling performance beyond what either one can accomplish alone.

According to the study in (Jeong, 2023), LLMs also offer solutions to complex logistical challenges, including scheduling and prioritisation, handling delays, resource allocation, contingency planning, and demand forecasting.

4.4. Engineering

Manufacturing equipment selection is essential in engineering because it directly affects automation design, enabling efficient production ramp-up, influencing production timelines, and overall operational efficiency. A work presented in (Werheid et al., 2024) proposes an LLM-based copilot to augment equipment selection decision-making, powered by retrieval-augmented generation (RAG) techniques. A multi-agent system is involved in this copilot, with each agent focusing on a specific part of the decision-making process. All these agents are coordinated by an orchestrator agent. For example, one agent selects an elementary operation (e.g. robotic handling), categorises the equipment type (e.g. robots, feeders), and identifies the specific equipment (e.g. a Cartesian Robot). Another agent evaluates the suitability of the selected equipment. If the selection is not suitable, the system prompts the user for further input.

The study in (Fan et al., 2025) presents an LLM-based framework for managing complex task parameters and adapting robotic capabilities in industrial manufacturing. To generate autonomous tool paths for complex spatial tasks, the framework uses LLM agents to match human-defined constraints with process parameters. GPT-3.5 and GPT-4.0 are used in the framework to simulate and interact with robotic environments to integrate embodied intelligence.

In additive manufacturing (AM), LLMs are also being used to support the creation of physical objects from digital models. AM provides design flexibility, product customisation, and reduces production costs by eliminating specialised tooling (Badini et al., 2023). As optimising G-code is a challenge in AM, several studies, such as (Badini et al., 2023; Eslaminia et al., 2024; Jignasu et al., 2024) show the efficacy of LLMs (such as GPT-4o, Claude 3.5 Sonnet, Llama-3.1-70B, Llama-3.1-405B, GPT-3.5, GPT-4, Bard, Claude-2, Llama-2-70B, and Starcoder) in G-code anomaly detection, troubleshooting, debugging, optimisation, and generation. Multi-modality capabilities of LLMs such as GPT-4V in AM are presented in (Picard et al., 2025). Picard et al., 2025 uses GPT-4V to identify the success of 3D-printing designs based on design for additive manufacturing (DfAM) and to output violations if the predicted design is incorrect.

CNC programming also leverages LLMs in subtractive manufacturing. CNC machines use G-codes to automate and control machining functions such as controlling tool movement, feed rate, spindle speed, and coolant flow. In addition to generating G-codes, CAM software needs to be customised or fine-tuned for specific operations. The detailed analysis in (Šket et al., 2024) shows ChatGPT's capabilities in generating and interpreting G-codes and detecting and simplifying errors produced by CAM software.

4.5. Quality

In the context of Industry 4.0, Zero Defect Manufacturing (ZDM) focuses on data-driven technologies to predict, prevent, and mitigate defects. To achieve ZDM, LLMs are being used for anomaly detection in industrial products ZDM (Deng et al., 2023; Liao et al., 2025; Wang & Dai, 2024).

An interoperable information modelling approach is introduced in (Shi et al., 2024a) to support quality control within ZDM, using Asset Administration Shell (AAS) and LLMs. AAS is an information modelling standard designed to carry structured data through its submodels that represent specific aspects of an asset (Shi et al., 2024a). In this approach, a quality control submodel is developed to align standardised vocabularies. To perform this semantic alignment, both pre-trained and fine-tuned LLMs are applied and evaluated to search model properties within standardised vocabulary repositories. With LLM-based semantic search, manual effort are reduced and quality control data are more consistent and interoperable, supporting ZDM goals.

A cause analysis is performed using LLMs to uncover complex causal relationships within quality data for aerospace product manufacturing (Zhou et al., 2024). A causal knowledge graph, called CausalKGPT, is developed in this work to facilitate reasoning and response generation for quality defects. A quality-related prompt dataset is constructed based on this knowledge graph. By using this dataset, a ChatGLM model is fine-tuned in a supervised process to capture complex causal relationships in aerospace quality data. In the study, CausalKGPT reduces observation bias and emphasises key quality factors to improve causal inference. CausalKGPT is more relevant to aerospace shell manufacturing than ChatGPT and GPT-4, based on experimental results.

4.6. Maintenance

Within Industry 4.0, log data comes from different resources, providing contextual and temporal information about events, anomalies, and operational states (Kohl et al., 2024; Le & Sahni, 2024; Liu et al., 2024c). Incorporating LLMs into log analysis enables contextual anomaly detection, proactive maintenance, and augments decision-making (Chandrasekaran & Perumal, 2025; Liu et al., 2024c; Sun et al., 2024).

In (Wang et al., 2024b), large vision and language models are used to optimise defect detection and maintenance decision-making. The introduced framework integrates domain-specific knowledge with a large vision model to analyse industrial images (e.g. images related to semiconductor manufacturing, infrastructure maintenance, and high-speed trains). Following this, an LLM with a knowledge base generates actionable maintenance recommendations and augments decision-making based on findings from the vision model.

In industrial maintenance, LLMs can support knowledge management and decision-making by integrating diverse data sources and domain-specific knowledge. A coal mine equipment maintenance dataset is created in (Cao et al., 2024) by collecting data from multiple sources-such as safety regulations, fault knowledge, operational instructions, and maintenance decision records. In the next step, the Low-Rank Adaptation technique is used to fine-tune ChatGLM based on the dataset. A knowledge graph is also constructed to link key aspects of coal mine equipment maintenance, including failure phenomena, fault causes, maintenance recommendations, and relevant subsystems. A fine-tuned model uses this knowledge graph to generate more relevant, accurate, and contextually informed maintenance responses.

4.7. Warehouse

To ensure the on-time delivery of finished products, warehousing operations must efficiently manage internal logistics. LLMs can help with tasks such as unloading, sorting, and docking. The research study introduced in (Kmieciak, 2025) integrates generic algorithms (GA) and LLM to optimise warehouse delivery scheduling in third-party logistics, increasing decision-making processes. In this approach, ChatGPT-4 is used to generate GA scripts via prompt engineering to optimise tasks such as unloading sequences and dock assignments. The feasibility of the approach is shown by results for a simplified warehouse scenario with three docks and six delivery trucks, each carrying different products. As a result of the GA scripts generated, the total length of travel to compile and distribute shipments is minimised, as well as the number of movements required.

The proposed multi-agent system is presented in (Quan & Liu, 2024) for inventory management in supply chains. Each stage of the supply chain, such as suppliers, warehouses, and retailers, is assigned an LLM-based agent that makes inventory decisions based on its local state. Managing state updates,

facilitating communication between agents, and interacting with the environment are all handled by a centralised user proxy. To guide LLM-based agents, the system applies chain-of-thought reasoning, resulting in more transparent, interpretable, and reliable decisions than traditional heuristics.

5. Discussion

This review contributes a phase-oriented, capability-driven perspective on the adoption of LLMs in manufacturing. Rather than cataloguing applications by industrial domain, the analysis maps core LLM capabilities to operational manufacturing phases, and positions use cases according to the strength of empirical evidence and evaluation maturity.

Based on the studies reviewed in Section 4, Figure 1 shows how key LLM capabilities align with the operational phases of manufacturing. This overall mapping highlights the current deployment focus of LLMs in manufacturing and indicates which capabilities are more prominent or remain underexplored. In documentation and decision-making, text and code understanding and generation abilities appear most frequently, while continuous learning is uncommon in practice. Advanced techniques such as continuous instruction tuning (CIT) and continuous model refinement (CMR) (Shi et al., 2024b; Wu et al., 2024) present promising approaches to enable continuous learning in manufacturing settings.

Within manufacturing, LLMs are not limited to automation tools; they increasingly augment human decision-making. In the face of demographic shifts, where experienced subject matter experts and seasoned engineers are retiring, there is a growing risk of losing critical tacit knowledge. This underscores the need for LLMs to bridge the expertise gap and promote human-centric manufacturing by capturing, preserving, and making accessible such domain-specific insights (Kernan Freire et al., 2024). Several LLM applications discussed in this review align with emerging Industry 5.0 principles by emphasising human-centric decision support and collaboration rather than fully autonomous automation. Industry 5.0 complements the Industry 4.0 paradigm by emphasising human-centricity, resilience, and sustainability alongside technological advancement, positioning LLMs as enablers of more adaptive manufacturing systems.

Figure 2 synthesises the reviewed literature by positioning LLM applications according to empirical evidence strength and evaluation maturity, presenting an imbalanced maturity landscape. Overall, most LLM applications in manufacturing are still at an exploratory or early validation stage. The majority of studies demonstrate conceptual feasibility and proof-of-concept implementations, but only a limited

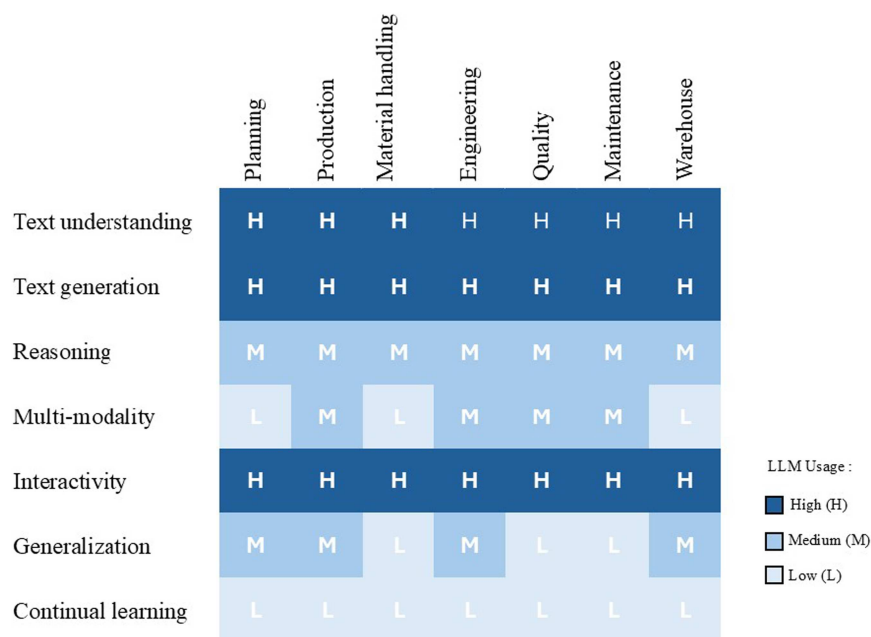


Figure 1. Heatmap of LLM capability coverage across manufacturing applications.

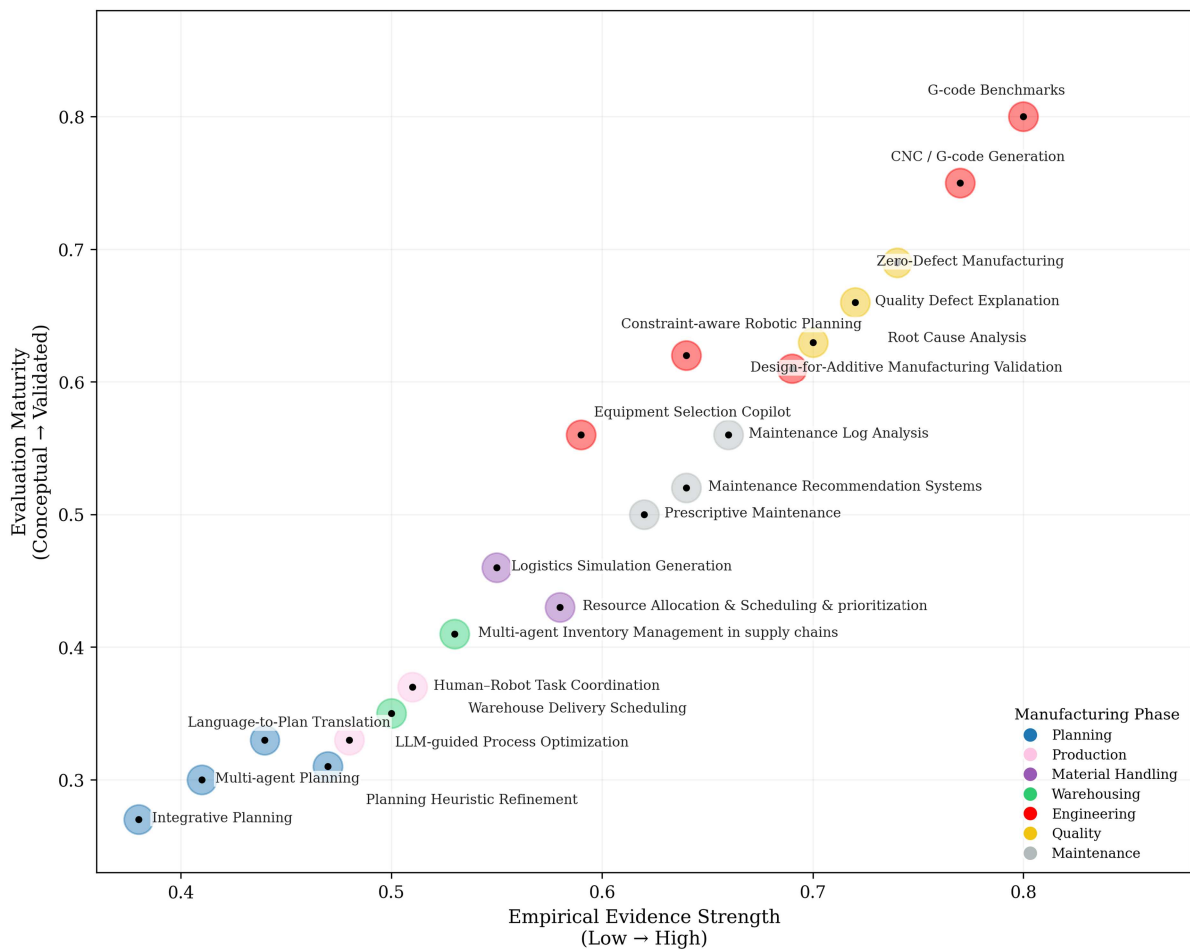


Figure 2. Empirical evidence and maturity of LLM applications across manufacturing phases.

subset offers empirical validation or systematic benchmarking. Applications in engineering and quality areas, such as CNC or G-code generation, zero-detect manufacturing, and quality defect analysis, exhibit comparatively higher levels of technological maturity and empirical evidence. Contrary to this, LLM use cases in planning remain exploratory, and evaluations involve simulations or small experiments.

Since LLMs like GPT-4, Claude, and Gemini are positioned to become standardised resources, innovation for most manufacturers will be driven by their own data and knowledge rather than the LLM technology itself. Consequently, a foundational step in leveraging LLMs is the collection and structuring of manufacturing data. As a result of this data, LLMs will perform better and be more accurate in manufacturing scenarios (Wevolver, 2024).

Existing studies often highlight successful LLM deployment in manufacturing scenarios; however, inaccurate or unreliable results are less frequently reported. Moreover, the majority of LLM-based prototypes have not yet been deployed in real-world manufacturing, underscoring the need for future-proof solutions (Wulf & Meierhofer, 2025). Contextually plausible but factually incorrect answers may be generated by LLMs, which misinterpret domain-specific terminology or fail to generalise to varying operational settings. Errors of this nature are particularly concerning in manufacturing environments where decisions directly affect safety, quality, and continuity of production.

Despite the diversity of LLM applications in industrial contexts, the reviewed studies show that LLMs differ from and complement traditional AI methods. Traditional AI approaches excel in well-defined problems that require precise and deterministic outputs, such as image segmentation or sensor-based anomaly detection. Factors such as data availability, model transparency, and scalability also influence which technology is most appropriate for a given task. LLMs, in contrast, are particularly effective in tasks

that require contextual understanding, content generation, knowledge integration, and adaptive recommendation. As a result, manufacturing benefits from both approaches depending on the specific problem at hand, and in many cases the two can be combined to form hybrid solutions.

The reviewed studies also reveal three main critical gaps that explain the uneven technology maturity of LLM applications in manufacturing and define clear research priorities for the future. First, adaptability is limited since no study has investigated continuous learning despite dynamic production conditions, revealing a mismatch between model assumptions and industrial reality. Furthermore, most applications lack generalisation across machines, plants, or product variants, thereby restricting industrial reuse. As a third issue, reliability is not adequately addressed, as safety-critical validation and uncertainty handling are rarely integrated.

6. Conclusion and outlook

The foundation language models, as text-driven technologies, are well-positioned for Industry 4.0. A review of the fundamental properties and capabilities of LLMs is provided in this paper, which discusses how they can be used to solve complex manufacturing problems. LLM capabilities are reviewed, addressing the first sub-research question by examining the technical characteristics that lead to their effectiveness in industrial and manufacturing environments. Following that, LLM applications are categorised according to operational phases, which addresses the second sub-research question by demonstrating how LLMs are applied across different stages of manufacturing.

A twofold limitation of this study can be identified. Due to rapid advances in AI and foundation language models, the findings and applications discussed may become outdated. Despite the detailed examination of capabilities and applications, domain-specific variations and contextual factors across manufacturing sectors are underemphasised, which may limit direct applicability in certain specialised environments. As shown in this paper, current LLM adoption is uneven across all capability dimensions. Although text understanding, text generation, and interactivity are consistently well represented across the manufacturing phases, more advanced properties, such as continual learning and generalisation, remain underdeveloped. Additionally, the reviewed studies demonstrate heterogeneous maturity across operational phases, with engineering and quality applications demonstrating higher levels of empirical validation than planning and material handling. These patterns highlight persistent limitations in adaptability, scalability, and safety-critical validation, which currently prevent LLM-based solutions from developing into fully reliable industrial systems.

In addition to these limitations, the deployment of LLMs in manufacturing environments comes with several broader challenges. Hallucination, data privacy, ethical and regulatory considerations, data deficiency, and modalities remain critical concerns, as industrial data often contains proprietary designs, production parameters, or trade secrets. Although domain-specific models have emerged, adapting these models to technical terms, structured sensor data, and specialised engineering documents remains a significant challenge. Furthermore, the lack of interpretability and transparency constrains the ability to understand the reasoning and decision-making processes of LLMs, while also limiting openness regarding their architectures and training data. Integrating LLMs into physical production environments and safety-critical manufacturing presents a situation with safety and reliability risks that must be carefully managed through robust validation, human oversight, and regulatory compliance.

Based on this study, future research should focus on systematically examining domain-specific adaptations of LLMs to better match the requirements, terminologies, and operational contexts of different manufacturing sectors. The results of these investigations can help identify current limitations and provide support for developing more tailored and effective solutions. Furthermore, future work can address the challenges associated with adopting foundation models, particularly in the collection and structuring of manufacturing data, which is crucial for enabling effective deployment and integration.

Author contributions

CRedit: **Sareh Aghaei**: Conceptualization, Formal analysis, Investigation, Methodology, Resources, Visualization, Writing – original draft, Writing – review & editing; **Fazel Ansari**: Conceptualization, Writing – review & editing.

Disclosure statement

The authors declare that they have no conflict of interest or financial conflicts to disclose.

Funding

This research was funded by Technische Universität Wien Bibliothek.

ORCID

Fazel Ansari  0000-0002-2705-0396

References

- Ahmad, W. U., Chakraborty, S., Ray, B., & Chang, K.-W. (2023). Summarize and generate to back-translate: Unsupervised translation of programming languages. In A. Vlachos, & I. Augenstein (Eds.), *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 1528–1542). Dubrovnik, Croatia: Association for Computational Linguistics. <https://doi.org/10.18653/v1/2023.eacl-main.112>
- Annepaka, Y., & Pakray, P. (2025). Large language models: A survey of their development, capabilities, and applications. *Knowledge and Information Systems*, 67(3), 2967–3022. <https://doi.org/10.1007/s10115-024-02310-4>
- Badini, S., Regondi, S., Frontoni, E., & Pugliese, R. (2023). Assessing the capabilities of chatgpt to improve additive manufacturing troubleshooting. *Advanced Industrial and Engineering Polymer Research*, 6(3), 278–287. <https://doi.org/10.1016/j.aiepr.2023.03.003>
- Barbosa, R., Santos, R., & Novais, P. (2025). Collaborative problem-solving with LLM: A multi-agent system approach to solve complex tasks using autogen. In G.-B. Alfonso, J. I. Vicente, E. B. Alia, M.-D. Cedric, J. Jaume, M. Karl, L. Fernando, & S. Nada (Eds.), *Highlights in Practical Applications of Agents, Multi-Agent Systems, and Digital Twins: The PAAMS Collection* (pp. 203–214). Cham: Springer Nature Switzerland.
- Baudin, M., & Netland, T. (2022). *Introduction to Manufacturing*. Routledge. New York.
- Bohao, P., Tian, Z., Liu, S., Yang, M.-C., & Jia, J. (2024). Scalable language model with generalized continual learning. *In the Twelfth International Conference on Learning Representations*.
- Cao, X., Xu, W., Zhao, J., Duan, Y., & Yang, X. (2024). Research on large language model for coal mine equipment maintenance based on multi-source text. *Applied Sciences*, 14(7), 2946. <https://doi.org/10.3390/app14072946>
- Chandrasekaran, B., & Perumal, V. (2025). Evaluation of few-shot learning with vision language models for needle bearing defect detection. *In SoutheastCon 2025* (pp. 1300–1308). <https://doi.org/10.1109/SoutheastCon56624.2025.10971495>
- Chang, T. A., & Bergen, B. K. (2024). Language model behavior: A comprehensive survey. *Computational Linguistics*, 50(1), 293–350. https://doi.org/10.1162/coli_a_00492
- Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., & Wang, Y. (2024). A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3), 1–45. <https://doi.org/10.1145/3641289>
- Chen, H., Wang, X., Zhou, Y., Huang, B., Zhang, Y., Feng, W., Chen, H., Zhang, Z., Tang, S., & Zhu, W. (2024b). Multi-modal generative AI: Multi-modal llmdiffusionbeyond. *arXiv preprint*.
- Chen, Z., Li, Y., & Wang, K. (2024a). Optimizing reasoning abilities in large language models: A step-by-step approach. *Authorea Preprints*.
- Deng, H., Guo, Y., Xu, Z., & Kang, Y. (2023). Ptmnet: Pixel-text matching network for zero-shot anomaly detection. *In 2023 9th International Conference on Big Data and Information Analytics (BigDIA)* (pp. 781–787). <https://doi.org/10.1109/BigDIA60676.2023.10429521>
- Ding, Y., Peng, J., Min, M. J., Kaiser, G., Yang, J., & Ray, B. (2024). Semcoder: Training code language models with comprehensive semantics reasoning. *arXiv preprint*.
- Dong, Q., Li, L., Dai, D., Zheng, C., Ma, J., Li, R., Xia, H., Xu, J., Wu, Z., Chang, B., Sun, X., Li, L., & Sui, Z. (2024). A survey on in-context learning. In Y. Al-Onaizan, M. Bansal, & Y.-N. Chen (Eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing* (pp. 1107–1128). Miami, Florida, USA: Association for Computational Linguistics. <https://doi.org/10.18653/v1/2024.emnlp-main.64>
- Eslaminia, A., Jackson, A., Tian, B., Stern, A., Gordon, H., Malhotra, R., Nahrstedt, K., & Shao, C. (2024). Fdm-bench: A comprehensive benchmark for evaluating large language models in additive manufacturing tasks. *arXiv preprint*.
- Fan, H., Liu, X., Fuh, J. Y. H., Lu, W. F., & Li, B. (2025). Embodied intelligence in manufacturing: Leveraging large language models for autonomous industrial robotics. *Journal of Intelligent Manufacturing*, 36(2), 1141–1157. <https://doi.org/10.1007/s10845-023-02294-y>
- Giallanza, A., La Scalia, G., Micale, R., & La Fata, C. M. (2024). Occupational health and safety issues in human-robot collaboration: State of the art and open challenges. *Safety Science*, 169, 106313. <https://doi.org/10.1016/j.ssci.2023.106313>
- Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., & Bi, X. (2025). Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint*.

- Hazra, R., Zuidberg Dos Martires, P., & De Raedt, L. (2024). Saycanpay: Heuristic planning with large language models using learnable domain knowledge, *In Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 38, pp. 20123–20133). <https://doi.org/10.1609/aaai.v38i18.29991>
- Ho, N., Schmid, L., & Yun, S.-Y. (2023). Large language models are reasoning teachers, *In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Long Papers)* (Vol. 1, pp. 14852–14882). Toronto, Canada: Association for Computational Linguistics. <https://doi.org/10.18653/v1/2023.acl-long.830>
- Huang, J., & Chang, K. C.- (2022). Towards reasoning in large language models: A survey. *arXiv preprint*
- Jackson, I., Jesus Saenz, M., & Ivanov, D. (2024). From natural language to simulations: Applying ai to automate simulation modelling of logistics systems. *International Journal of Production Research*, 62(4), 1434–1457. <https://doi.org/10.1080/00207543.2023.2276811>
- Jeong, Y. (2023). Digitalization in production logistics: How ai, digital twins, and simulation are driving the shift from model-based to data-driven approaches. *International Journal of Precision Engineering and Manufacturing-Smart Technology*, 1, 187–200. <https://doi.org/10.57062/ijpem-st.2023.0052>
- Jiang, J., Wang, F., Shen, J., Kim, S., & Kim, S. (2024). A survey on large language models for code generation. *arXiv preprint*.
- Jignasu, A., Marshall, K., Ganapathysubramanian, B., Balu, A., Hegde, C., & Krishnamurthy, A. (2024). Evaluating large language models for g-code debugging, manipulation, and comprehension, *In 2024 IEEE LLM Aided Design Workshop (LAD)* (pp. 1–5). IEEE. <https://doi.org/10.1109/LAD62341.2024.10691700>
- Keele, S. (2007). Guidelines for performing systematic literature reviews in software 681 engineering. Technical report, Technical report, ver. 2.3 ebse technical report. 682 ebse.
- Kernan Freire, S., Wang, C., Foosherian, M., Wellsandt, S., Ruiz-Arenas, S., & Niforatos, E. (2024). Knowledge sharing in manufacturing using llm-powered tools: User study and model benchmarking. *Frontiers in Artificial Intelligence*, 7, 1293084. <https://doi.org/10.3389/frai.2024.1293084>
- Kmiecik, M. (2025). Creating a genetic algorithm for third-party logistics' warehouse delivery scheduling via a large language model. *Journal of Modelling in Management*, 20, 1138–1162. <https://doi.org/10.1108/JM2-06-2024-0192>
- Kohl, L., Eschenbacher, S., Besinger, P., & Ansari, F. (2024). Large language model-based chatbot for improving human-centricity in maintenance planning and operations, *In Proceedings of the European Conference of the Prognostics and Health Management Society 2024* (Vol. 8, p. 12). <https://doi.org/10.36001/phme.2024.v8i1.4098>
- Konstantinou, C., Antonarakos, D., Angelakis, P., Gkournelos, C., Michalos, G., & Makris, S. (2024). Leveraging generative ai prompt programming for human-robot collaborative assembly. *Procedia CIRP*, 128, 621–626. <https://doi.org/10.1016/j.procir.2024.03.040>
- Kumar, P. (2024). Large language models (llms): Survey, technical frameworks, and future challenges. *Artificial Intelligence Review*, 57(10), 260. <https://doi.org/10.1007/s10462-024-10888-y>
- Le, E., & Sahni, H. (2024). Utilizing large language models for logged data analysis in industrial contexts: Investigating the adaptation of large language models for logged data analysis. Master's thesis, Chalmers University of Technology, Department of Computer Science and Engineering.
- Leung, C. S. K., & Lau, H. Y. K. (2020). Simulation-based optimization for material handling systems in manufacturing and distribution industries. *Wireless Networks*, 26(7), 4839–4860. <https://doi.org/10.1007/s11276-018-1894-x>
- Li, J., Tang, T., Zhao, W. X., Nie, J. -Y., & Wen, J. -R. (2024a). Pre-trained language models for text generation: A survey. *ACM Computing Surveys*, 56(9), 1–39.
- Li, J., Yang, Y., Bai, Y., Zhou, X., Li, Y., Sun, H., Liu, Y., Si, X., Ye, Y., & Wu, Y. (2024b). Fundamental capabilities of large language models and their applications in domain scenarios: A survey, *In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Long Papers)* (Vol. 1, pp. 11116–11141).
- Li, M., Zhao, S., Wang, Q., Wang, K., Zhou, Y., Srivastava, S., Gokmen, C., Lee, T., Li, E. L., & Zhang, R. (2024c). Embodied agent interface: Benchmarking llms for embodied decision making. *Advances in Neural Information Processing Systems*, 37, 100428–100534.
- Li, S., Puig, X., Paxton, C., Du, Y., Wang, C., Fan, L., Chen, T., Huang, D. -A., Akyürek, E., Anandkumar, A., Andreas, J., Mordatch, I., Torralba, A., & Zhu, Y. (2022). Pre-trained language models for interactive decision-making, *In Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*. Red Hook, NY, USA: Curran Associates Inc.
- Li, S., Wang, Z., Yan, Z., Gao, Y., Jiang, H., & Zheng, P. (2024d). Large language model for humanoid cognition in proactive human-robot collaboration, *In 2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)* (pp. 540–545). <https://doi.org/10.1109/CASE59546.2024.10711379>
- Li, X., Wang, S., Sun, J., & Xu, Z. (2023). Variational data-free knowledge distillation for continual learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10), 12618–12634. <https://doi.org/10.1109/TPAMI.2023.3271626>
- Liao, P., Chien, P. -C., Tsukida, H., Kato, Y., & Ohya, J. (2025). FFAD: Fixed-position few-shot anomaly detection for wire harness utilizing vision-language models, *In Proceedings of the 14th International Conference on Pattern Recognition Applications and Methods* (pp. 647–656). SciTePress. <https://doi.org/10.5220/0013164400003905>
- Lim, J., Patel, S., Evans, A., Pimley, J., Li, Y., & Kovalenko, I. (2024). Enhancing human-robot collaborative assembly in manufacturing systems using large language models, *2024 IEEE 20th International Conference on Automation Science and Engineering* (pp. 2581–2587). IEEE. <https://doi.org/10.1109/CASE59546.2024.10711843>

- Liu, A., Feng, B., Xue, B., Wang, B., Wu, B., Lu, C., Zhao, C., Deng, C., Zhang, C., & Ruan, C. (2024b). Deepseek-v3 technical report. *arXiv preprint*, <https://doi.org/10.48550/arXiv.2412.19437>
- Liu, F., Liu, Y., Shi, L., Huang, H., Wang, R., Yang, Z., & Zhang, L. (2024a). Exploring and evaluating hallucinations in llm-powered code generation. *arXiv preprint*, <https://doi.org/10.48550/arXiv.2404.00971>
- Liu, Y., Ji, Y., Tao, S., He, M., Meng, W., Zhang, S., Yongqian, S., Xie, Y., Chen, B., & Yang, H. (2024c). Loglm: From task-based to instruction-based automated log analysis. *arXiv preprint*, <https://doi.org/10.48550/arXiv.2410.09352>
- Matarazzo, A., & Torlone, R. (2025). A survey on large language models with some insights on their capabilities and limitations. <https://doi.org/10.48550/ARXIV.2501.04040>
- Mondal, S. S., Webb, T. W., Wang, C., Krabach, B., & Momennejad, I. (2024). A prefrontal cortex-inspired architecture for planning in large language models. <https://openreview.net/forum?id=SkETBJRKH7>
- Mu, F., Shi, L., Wang, S., Yu, Z., Zhang, B., Wang, C., Liu, S., & Wang, Q. (2024). Clarifygpt: A framework for enhancing llm-based code generation via requirements clarification. *Proceedings of the ACM on Software Engineering*, 1(FSE), 2332–2354. <https://doi.org/10.1145/3660810>
- Naveed, H., Khan, A. U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Akhtar, N., Barnes, N., & Mian, A. (2025). A comprehensive overview of large language models. *ACM Transactions on Intelligent Systems and Technology*, 16(5), <https://doi.org/10.1145/3744746>
- Nijkamp, E., Hayashi, H., Xiong, C., Savarese, S., & Zhou, Y. (2023). Codegen2: Lessons for training llms on programming and natural languages. *arXiv preprint*, <https://doi.org/10.48550/arXiv.2305.02309>
- Pallagani, V., Muppasani, B. C., Roy, K., Fabiano, F., Loreggia, A., Murugesan, K., Srivastava, B., Rossi, F., Horesh, L., & Sheth, A. P. (2024). On the prospects of incorporating large lan780 guage models (LLMs) in automated planning and scheduling (APS). In *34th 781 International Conference on Automated Planning and Scheduling* (Vol. P, pp. 432–444). <https://openreview.net/forum?id=BLsvMLvuhL>. <https://doi.org/10.1609/icaps.v34i1.31503>
- Picard, C., Edwards, K. M., Doris, A. C., Man, B., Giannone, G., Alam, M. F., & Ahmed, F. (2025). From concept to manufacturing: Evaluating vision-language models for engineering design. *Artificial Intelligence Review*, 58(9), 288. <https://doi.org/10.1007/s10462-025-11290-y>
- Quan, Y., & Liu, Z. (2024). Invagent: A large language model based multi-agent system for inventory management in supply chains. *arXiv preprint*.
- Radford, A., & Narasimhan, K. (2018). Improving language understanding by generative pre-training. Technical report.
- Rastogi, A., Jiang, A. Q., Lo, A., Berrada, G., Lample, G., Rute, J., Barmentlo, J., Yadav, K., Khandelwal, K., & Chandu, K. R. (2025). Magistral. *arXiv preprint*, <https://arxiv.org/abs/2506.10910>
- Sermanet, P., Ding, T., Zhao, J., Xia, F., Dwibedi, D., Gopalakrishnan, K., Chan, C., Dulac-Arnold, G., Maddineni, S., Joshi, N. J., Florence, P., Han, W., Baruch, R., Lu, Y., Mirchandani, S., Xu, P., Sanketi, P. R., Hausman, K., Shafran, I., ... Cao, Y. (2023). Robovqa: Multimodal long-horizon reasoning for robotics. In *2024 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 645–652). <https://api.semanticscholar.org/CorpusID:264935351>.
- Shi, D., Liedl, P., & Bauernhansl, T. (2024a). Interoperable information modelling leveraging asset administration shell and large language model for quality control toward zero defect manufacturing. *Journal of Manufacturing Systems*, 77, 678–696. <https://doi.org/10.1016/j.jmsy.2024.10.011>
- Shi, H., Xu, Z., Wang, H., Qin, W., Wang, W., Wang, Y., Wang, Z., Ebrahimi, S., & Wang, H. (2024b). Continual learning of large language models: A comprehensive survey. *ACM Computing Surveys*, 58(5), 1–42. <https://doi.org/10.1145/3735633>
- Šket, K., Potočnik, D., Ficko, M., & Klančnik, S. (2024). Enhancing g-code programming in cnc machining using chatgpt: A comparative study of gpt-3.5 and gpt-4.0. *SSRN Electronic Journal*, <https://doi.org/10.2139/ssrn.4940034>
- Sun, Q., Li, Y., Zhou, C., & Tian, Y.-C. (2024). Root cause analysis for industrial process anomalies through the integration of knowledge graph and large language model. In *2024 43rd Chinese Control Conference (CCC)* (pp. 6855–6860). <https://doi.org/10.23919/CCC63176.2024.10662704>
- Tamkin, A., Brundage, M., Clark, J., & Ganguli, D. (2021). Understanding the capabilities, limitations, and societal impact of large language models. *arXiv preprint*.
- Tsushima, Y., Yamamoto, S., Ravankar, A. A., Luces, J. V. S., & Hirata, Y. (2025). Task planning for a factory robot using large language model. *IEEE Robotics and Automation Letters*, 10(3), 2383–2390. <https://doi.org/10.1109/LRA.2025.3531153>
- Urlana, A., Kumar, C., Singh, A., Garlapati, B. M., Chalamala, S., & Mishra, R. (2024). Llms with industrial lens: Deciphering the challenges and prospects -a survey.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.
- Vogel-Heuser, B., & Hess, D. (2016). Guest editorial industry 4.0–prerequisites and visions. *IEEE Transactions on Automation Science and Engineering*, 13(2), 411–413. <https://doi.org/10.1109/TASE.2016.2523639>
- Wang, H., Li, C., Li, Y. -F., & Tsung, F. (2024b). An intelligent industrial visual monitoring and maintenance framework empowered by large-scale visual and language models. *IEEE Transactions on Industrial Cyber-Physical Systems*, 2, 166–175. <https://doi.org/10.1109/TICPS.2024.3414292>

- Wang, T., Fan, J., & Zheng, P. (2024a). An llm-based vision and language cobot navigation approach for human-centric smart manufacturing. *Journal of Manufacturing Systems*, 75, 299–305. <https://doi.org/10.1016/j.jmsy.2024.04.020>
- Wang, Z., & Dai, L. (2024). Leveraging large language model for robust industrial image anomaly detection, *In 2024 6th International Conference on Frontier Technologies of Information and Computer (ICFTIC)* (pp. 671–674). <https://doi.org/10.1109/ICFTIC64248.2024.10912957>
- Wang, Z., & Qin, H. (2024). Intelligent industrial production process automatic regulation system based on LLM agents, *In 2024 5th International Conference on Artificial Intelligence and Electromechanical Automation (AIEA)* (pp. 133–137). IEEE. <https://doi.org/10.1109/AIEA62095.2024.10692701>
- Wang, Z., Zhang, G., Yang, K., Shi, N., Zhou, W., Hao, S., Xiong, G., Li, Y., Yuan Sim, M., & Chen, X. (2023). Interactive natural language processing. *arXiv preprint*, <https://arxiv.org/abs/2305.13246>
- Werheid, J., Melnychuk, O., Zhou, H., Huber, M., Rippe, C., Joosten, D., Keskin, Z., Wittstamm, M., Subramani, S., Drescher, B., Göppert, A., Abdelrazeq, A., & Schmitt, R. (2024). Designing an LLM-based copilot for manufacturing equipment selection. *arXiv preprint*, <https://doi.org/10.48550/arXiv.2412.13774>
- Wevolver. (2024). Shaping the future of manufacturing with llms and generative ai. Accessed: 2025-06-26. URL. <https://www.wevolver.com/article/shaping-the-future-of-manufacturing-with-llms-and-generative-ai>
- Wu, T., Luo, L., Li, Y., -F., Pan, S., Vu, T., -T., & Haffari, G. (2024). Continual learning for large language models: A survey. *arXiv preprint*, <https://arxiv.org/abs/2402.01364>
- Wulf, J., & Meierhofer, J. (2025). Mitigating risks in large language model applications for manufacturing. <https://digitalcollection.zhaw.ch/handle/11475/33327>
- Xu, B., Liu, X., Shen, H., Han, Z., Li, Y., Yue, M., Peng, Z., Liu, Y., Yao, Z., & Xu, D. (2023). Gentopia.AI: A collaborative platform for tool-augmented LLMs. In Y. Feng, & E. Lefever (Eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (pp. 237–245). Singapore: Association for Computational Linguistics. <https://doi.org/10.18653/v1/2023.emnlp-demo.20>
- Xu, F., Hao, Q., Zong, Z., Wang, J., Zhang, Y., Wang, J., Lan, X., Gong, J., Ouyang, T., & Meng, F. (2025). Towards large reasoning models: A survey of reinforced reasoning with large language models. *arXiv preprint*, <https://arxiv.org/abs/2501.09686>
- Yang, Y., Zhou, J., Ding, X., Huai, T., Liu, S., Chen, Q., Xie, Y., & He, L. (2024). Recent advances of foundation language models-based continual learning: A survey. *ACM Computing Surveys*, 57(5), 1–38. <https://doi.org/10.1145/3705725>
- Zhang, C., Chen, Y., Chen, H., & Chong, D. (2024a). Industry 4.0 and its implementation: A review. *Information Systems Frontiers*, 26(5), 1773–1783. <https://doi.org/10.1007/s10796-021-10153-5>
- Zhang, H., Du, W., Shan, J., Zhou, Q., Du, Y., Tenenbaum, J. B., Shu, T., & Gan, C. (2024b). Building cooperative embodied agents modularly with large language models, *In The Twelfth International Conference on Learning Representations*. <https://openreview.net/forum?id=EnXJfQqy0K>.
- Zheng, J., Qiu, S., Shi, C., & Ma, Q. (2024). Towards lifelong learning of large language models: A survey. *arXiv preprint*, <https://arxiv.org/abs/2406.06391>
- Zhou, B., Li, X., Liu, T., Xu, K., Liu, W., & Bao, J. (2024). Causalkgpt: Industrial structure causal knowledge-enhanced large language model for cause analysis of quality problems in aerospace product manufacturing. *Advanced Engineering Informatics*, 59, 102333. <https://doi.org/10.1016/j.aei.2023.102333>