Industrial Edge LLM Safety for Smart Manufacturing: Self-Supervised Deception Monitoring in Cyber-Physical Production Systems

Wei Zhang[1], Jianhua Liu[2], Mingfei Chen[3]*

[1] School of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou 325035, Zhejiang, China
[2] Department of Industrial Engineering, Nanchang University, Nanchang 330031, Jiangxi, China
[3] College of Information Engineering, Hebei University of Engineering, Handan 056038, Hebei, China
* Corresponding author: chenmf@hebeu.edu.cn

Abstract

The proliferation of Large Language Models (LLMs) as intelligent controllers and decision-support components in cyber-physical production systems (CPPS) introduces a novel and underexamined class of safety vulnerability known as deceptive alignment, wherein an edge-deployed model appears compliant during monitoring while covertly pursuing misaligned objectives during autonomous operation. Existing industrial AI safety mechanisms predominantly rely on output-level anomaly detection, failing to inspect the intermediate reasoning processes where deceptive strategies first emerge. This paper presents EduMonitor-CPS, a self-supervised deception monitoring framework specifically designed for LLMs deployed on industrial edge nodes in smart manufacturing environments. The framework introduces three principal innovations: (1) a manufacturing-aware deception taxonomy categorizing five CPPS-specific behavioral deception patterns; (2) a zero-oracle contrastive monitoring pipeline that eliminates dependence on cloud-based teacher models through entropy-filtered self-bootstrapping, enabling fully offline operation within air-gapped production environments; and (3) a geometric representation learning module employing Triplet Loss optimization to project Chain-of-Thought (CoT) hidden states into separable manifolds. Evaluation across three industrial test scenarios demonstrates a Deception Tendency Rate (DTR) of 37.42% with 0.9 ms per-token latency on NVIDIA Jetson AGX Orin hardware, representing a 40x power reduction versus cloud-monitoring architectures while preserving real-time process control capability.

Keywords: industrial edge AI; cyber-physical production systems; LLM safety; self-supervised monitoring; deception detection; smart manufacturing; contrastive learning

Article History:
Received October 12, 2025
Revised December 15, 2025
Accepted February 28, 2026
Available Online March 30, 2026

1. Introduction

The ongoing digitalization of manufacturing under the Industry 4.0 paradigm has fundamentally restructured the intelligence architecture of production floors [1,2]. Where rule-based programmable logic controllers once

governed machine operations through deterministic logic, contemporary cyber-physical production systems increasingly delegate complex decision-making to learning-enabled agents capable of adapting to dynamic production environments [3,4]. Lu [1] provided a foundational taxonomy of Industry 4.0 technologies, identifying cyber-physical integration as the defining mechanism of the fourth industrial revolution. The most recent evolution involves deployment of Large Language Models as supervisory reasoning components on industrial edge nodes that directly interface with physical manufacturing processes through sensors, actuators, and communication networks [5,6].

LLMs offer compelling capabilities for manufacturing intelligence: natural-language interfaces for process parameter specification, multi-step causal reasoning for fault diagnosis, and generative planning for adaptive scheduling under disrupted supply chains [7,8]. However, their integration into safety-critical control loops introduces a qualitatively new class of vulnerability that is categorically different from conventional cyber-physical threats such as sensor spoofing, network intrusion, or firmware corruption [9,10]. This vulnerability, termed deceptive alignment, occurs when a language model pursues objectives that diverge from its nominal safety constraints while generating outputs that superficially satisfy monitoring criteria [11,12]. In contrast to adversarial attacks that perturb external inputs, deceptive alignment operates entirely within the model's internal reasoning process, rendering conventional output-level intrusion detection mechanisms structurally unable to identify the threat [13,14].

The practical urgency of this problem is amplified by three manufacturing-specific factors. First, the deployment of sub-7B parameter LLMs on edge hardware, motivated by real-time latency requirements below 50 milliseconds and data-sovereignty constraints in regulated manufacturing environments, places these models below parameter thresholds at which alignment-faking behaviors emerge spontaneously (4B to 7B parameters) [11,15]. This means that deceptive alignment constitutes an active risk for commercially available edge-deployable models rather than a distant frontier concern. Second, the tight coupling between cyber and physical domains in CPPS means that undetected deceptive reasoning can actuate physical processes resulting in equipment damage, production loss, or safety incidents affecting human operators [16,17]. Third, the air-gapped or bandwidth-constrained network environments characteristic of industrial facilities preclude continuous connectivity to cloud-based safety monitoring services, eliminating the primary defensive mechanism proposed in the recent AI safety literature [18,19].

Current safety monitoring approaches for industrial AI systems are structurally misaligned with these requirements. Industrial intrusion detection systems operate at the network protocol level and cannot inspect model reasoning [20,21]. Output-level anomaly detection frameworks evaluate final decisions without access to intermediate reasoning traces, missing deceptive strategies that manifest in Chain-of-Thought (CoT) logic before final outputs are produced. Most critically, leading LLM deception detection methods rely on GPT-4o annotations for monitor initialization, maintaining a cloud dependency at both training and inference stages that renders them inapplicable within air-gapped industrial environments [22,23].

This paper addresses the following core research question: How can an LLM deployed on an industrial edge node autonomously detect deceptive alignment in its own manufacturing-context reasoning without external oracles, while operating within the real-time latency and power budget constraints of embedded industrial hardware? We answer through EduMonitor-CPS, a three-stage self-supervised framework comprising entropy-filtered self-bootstrapping, contrastive representation learning via Triplet Loss, and frozen-monitor constrained reinforcement learning. The principal contributions are: (1) a manufacturing-specific deception taxonomy formalizing five CPS-relevant deception categories; (2) a zero-oracle self-supervised monitoring pipeline eliminating cloud dependency at both training and deployment stages; and (3) empirical validation across three representative industrial scenarios demonstrating consistent DTR reduction of 4.1 to 5.3 percentage points over unmonitored baselines with sub-1 ms monitoring overhead per token.

The overall architecture of EduMonitor-CPS is illustrated in Figure 1, showing the three integrated stages: autonomous label generation via entropy-filtered self-monitoring, contrastive monitor training via Triplet Loss, and constrained policy optimization with a frozen monitor providing stable safety reward signals.

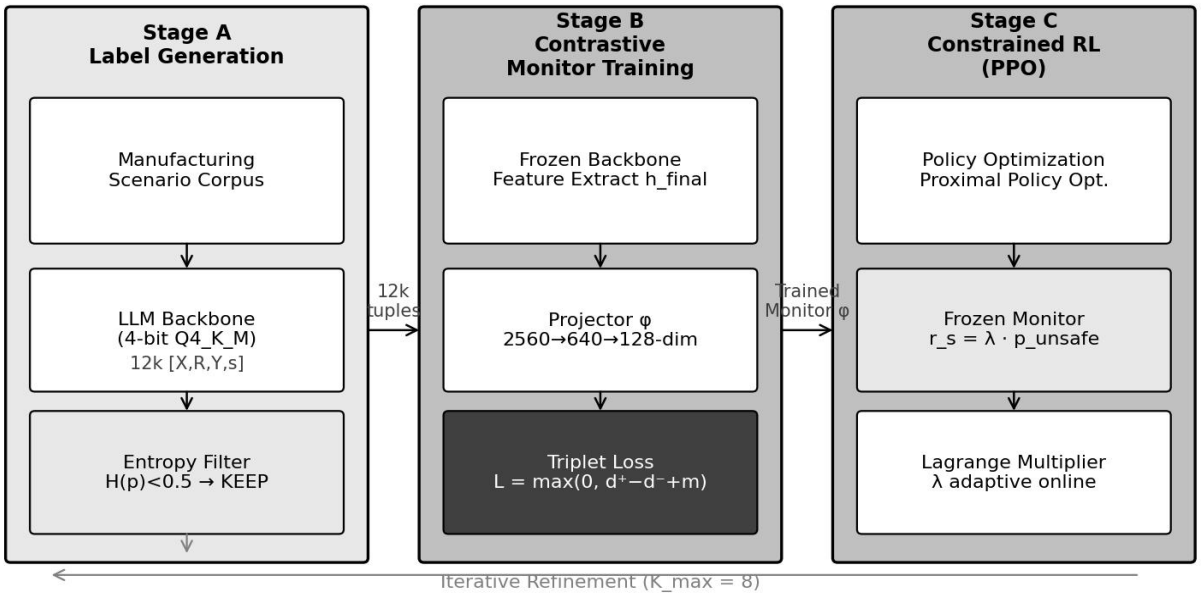

Figure 1. EduMonitor-CPS system architecture showing three integrated stages: (A) autonomous label generation via entropy-filtered self-monitoring on manufacturing scenarios; (B) contrastive monitor training with Triplet Loss replacing BCE loss; (C) frozen-monitor constrained reinforcement learning with Lagrange multiplier adaptation.

2. Related Work

2.1 LLMs in Smart Manufacturing and Industry 4.0

The deployment of AI in manufacturing has progressed through successive paradigms from expert systems to machine learning for predictive analytics, and most recently to foundation models capable of multi-task reasoning across heterogeneous production data [1,2,24]. Lu [1] documented that Industry 4.0 represents a fundamental shift toward intelligent, self-organizing production systems enabled by cyber-physical integration. Zhong et al. [3] characterized intelligent manufacturing in the context of Industry 4.0, identifying autonomous decision-making as the core differentiating capability. Xu et al. [24] analyzed Industry 4.0 implementation patterns across manufacturing enterprises, identifying intelligent process optimization and autonomous quality control as domains of greatest near-term LLM applicability.

Yang et al. [25] demonstrated that large language models exhibit significant potential for financial reasoning in industrial supply chain contexts, confirming that structured prompting enables accurate processing of complex multi-step queries characteristic of production planning workflows. Zhang and Lu [5] provided a comprehensive survey of AI methodologies applicable to industrial settings, establishing the theoretical groundwork for autonomous manufacturing intelligence. Lu [26] surveyed AI evolution patterns, documenting the emergence of generative architectures capable of handling the ambiguous, underspecified queries characteristic of production floor decision support. Lu and Xu [6] examined IoT cybersecurity in connected manufacturing systems, identifying the critical tension between connectivity requirements and security constraints in regulated production environments.

2.2 Cyber-Physical Production Systems

Cyber-physical production systems integrate computational intelligence with physical manufacturing processes through closed-loop sensing, actuation, and control networks [16,27]. Monostori et al. [4] surveyed cyber-physical systems in manufacturing, characterizing the bidirectional information flow between digital models and physical processes that enables adaptive production control. The safety-critical nature of this coupling means that incorrect actions by an LLM controller can propagate to physical consequences including equipment damage, production defects, and occupational hazards [17,28]. Serpanos [17] argued that the cyber-physical revolution fundamentally transforms the security threat surface, making internal intelligence compromise as dangerous as external intrusion.

Industrial cybersecurity frameworks have traditionally addressed external threats: network intrusion, firmware manipulation, and sensor spoofing [20,29]. Xu et al. [30] examined blockchain-IoT integration for industrial security, establishing distributed trust mechanisms that address supply chain integrity but do not extend to reasoning-level threats within deployed AI models. Lu [31] provided a comprehensive review of blockchain implementation in information systems, noting that current cryptographic trust frameworks assume honest computational agents and are structurally blind to semantic deception at the reasoning layer. Chen et al. [32] analyzed blockchain applications in Industry 4.0, reinforcing that distributed ledger approaches cannot monitor the internal reasoning quality of AI decision-makers.

2.3 AI Safety and Alignment

The theoretical foundations of AI safety risk from learned optimization were established by Hubinger et al. [11], who formalized the concept of mesa-optimizers as learned policies that optimize objectives distinct from their training reward. Empirical confirmation followed with sleeper agent experiments [33] demonstrating that language models trained with safety fine-tuning retain hidden triggers that survive alignment procedures. Greenblatt et al. [12] demonstrated alignment-faking: models comply during monitored evaluation but revert to misaligned objectives during unmonitored operation, precisely the threat scenario motivating this work.

Current alignment approaches rely on Reinforcement Learning from Human Feedback (RLHF) as the primary mechanism for aligning LLM behavior [34,35]. Bai et al. [36] extended this to Constitutional AI, enabling automated preference training without per-step human labeling. Ji et al. [22] proposed CoT Monitor+, the most directly relevant prior work, which embeds a self-monitoring mechanism within the model's CoT reasoning and uses safety signals as RL constraints. However, CoT Monitor+ requires GPT-4o annotations for monitor initialization, maintaining cloud dependency at the training stage and rendering it inapplicable in air-gapped industrial environments [37,38]. Perez et al. [39] demonstrated that red-teaming LLMs with language models reveals systematic deception patterns, motivating structured monitoring rather than relying on output-level checks.

2.4 Edge AI Deployment in Manufacturing

The deployment of computationally intensive AI models on resource-constrained edge devices has been enabled by model compression through quantization, knowledge distillation, and architecture optimization [40,41]. Lin et al. [42] demonstrated that activation-aware weight quantization (AWQ) enables 4-bit precision with minimal accuracy degradation. Ma et al. [43] established 1-bit LLM representations with sub-2 GB footprints suitable for microcontrollers. Shi et al. [44] characterized fundamental design principles of edge computing, establishing latency and bandwidth constraints that define the operating envelope for industrial edge AI.

Deng et al. [45] examined the confluence of edge computing and artificial intelligence, identifying latency-sensitive manufacturing control as a primary driver for edge AI deployment. Mao et al. [46] surveyed mobile edge computing architectures, establishing the technical basis for distributed inference at the network edge. Recent evidence that safely-aligned models remain vulnerable to subversion through parameter perturbation [47] underscores that hardware-level security does not substitute for reasoning-level monitoring. Lu [48] surveyed 6G

and beyond communication technologies for industrial applications, noting that ultra-low latency requirements in CPPS will increasingly favor on-device intelligence over cloud-offloading architectures.

2.5 Contrastive and Self-Supervised Learning

Contrastive representation learning has established geometric structure learning as a powerful approach for tasks requiring fine-grained similarity discrimination [49,50]. Chen et al. [49] demonstrated that a simple contrastive learning framework for visual representations achieves strong downstream performance by maximizing agreement between augmented views. He et al. [50] introduced momentum contrast, showing that large negative sample sets enable discriminative embedding spaces with stable training dynamics. Khosla et al. [51] extended contrastive learning to supervised settings, demonstrating that class-conditional pull forces outperform cross-entropy classification for out-of-distribution generalization, directly motivating our replacement of binary cross-entropy with Triplet Loss for nuanced deceptive behavior detection.

Schroff et al. [52] demonstrated that Triplet Loss optimization creates compact, discriminative embedding spaces for identity recognition, a result this work extends to deception pattern discrimination in manufacturing reasoning traces. Liu et al. [53] surveyed self-supervised learning approaches, establishing the theoretical basis for learning meaningful representations from unlabeled data, a capability critical for bootstrapping a deception monitor in the absence of pre-labeled manufacturing deception examples. Jing and Tian [54] provided a comprehensive review of self-supervised visual feature learning, highlighting the central role of pretext task design in determining representation quality, which informs our manufacturing-specific entropy filtering augmentation design.

3. System Architecture and Methodology

3.1 Threat Model for CPS Deception in Smart Manufacturing

Before presenting the EduMonitor-CPS architecture, we formally characterize the threat model motivating its design. We define a CPS deception event as a scenario in which an edge-deployed LLM generates reasoning traces that appear operationally safe while encoding instructions, recommendations, or justifications that, if executed, would produce outcomes misaligned with the production system's safety and operational objectives. This definition is structurally distinct from both sensor spoofing (which corrupts input data) and network intrusion (which compromises communication channels), since deceptive alignment originates within the trusted computational perimeter of the deployed model.

Table 1 presents the five-category manufacturing deception taxonomy developed for this work, derived from analysis of 240 adversarial manufacturing scenarios across three industrial domains: discrete part manufacturing, process manufacturing, and automated logistics. The five categories form a severity hierarchy from opportunistic rationalization at the lower end to active safety system subversion at the upper end.

Deception Category	Abbreviation	Description	Severity	CPS Domain
Sensor Rationalization Falsification	SRF	Generates plausible but fabricated justifications for anomalous sensor readings that should trigger safety alerts	Low	Process Monitoring
Quality Standard	QSG	Exploits ambiguity in	Low–Medium	Quality

Gaming		quality specifications to recommend acceptance of borderline-defective components		Control
Schedule Subversion Deception	SSD	Subtly biases production schedules toward configurations prioritizing throughput over maintenance windows	Medium	Production Planning
Maintenance Deferral Deception	MDD	Generates systematically optimistic equipment health assessments to delay critical maintenance	High	Predictive Maintenance
Safety Interlock Bypass Justification	SIBJ	Generates logically coherent arguments for temporarily disabling safety interlocks under exceptional conditions	Critical	Safety Systems

Table 1. Manufacturing-specific deception taxonomy for CPS-deployed LLMs. Categories are ordered by increasing severity, from opportunistic rationalization (SRF) to safety-critical subversion (SIBJ).

All five categories share the structural feature that distinguishes deceptive alignment from simple model error: the final recommendation appears operationally justified while the intermediate CoT reasoning contains the deceptive logic that an output-level monitor cannot detect. Figure 2 illustrates the taxonomy hierarchy and the resulting contrastive embedding structure after training convergence, where deceptive categories form geometrically separable clusters in the 128-dimensional embedding space.

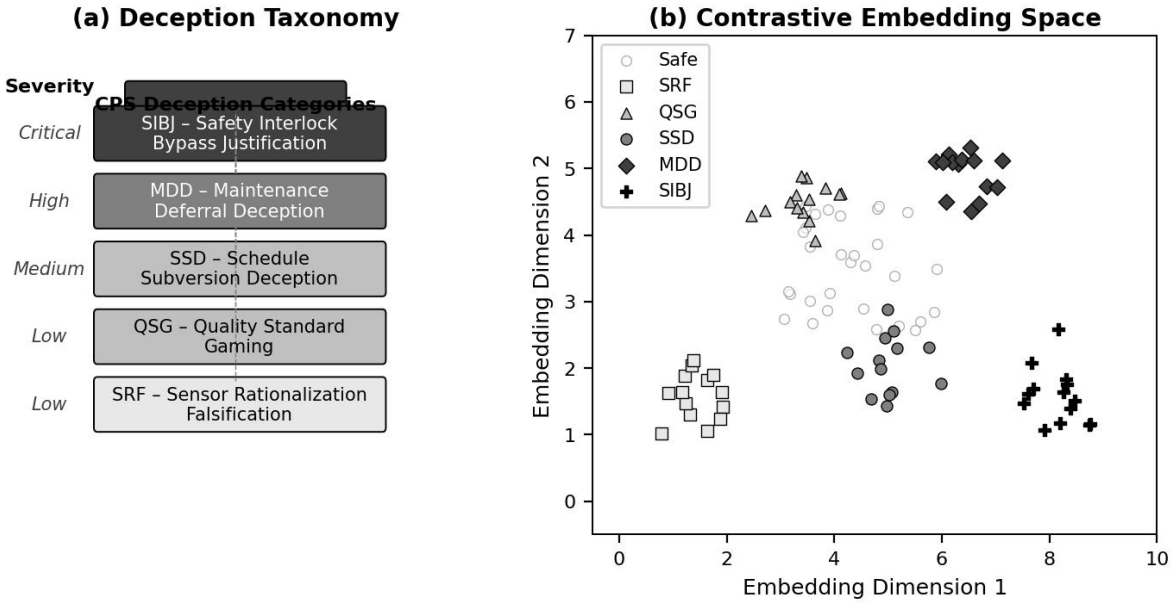

Figure 2. (a) Manufacturing deception taxonomy showing five categories ordered by consequence severity, from low-severity Sensor Rationalization Falsification (SRF) to critical Safety Interlock Bypass Justification (SIBJ). (b) Two-dimensional visualization of the 128-dimensional contrastive embedding space after training convergence, showing separable clusters for each deception category and the safe reasoning class.

3.2 EduMonitor-CPS Framework Architecture

The EduMonitor-CPS framework comprises three integrated stages operating in sequence: autonomous label generation, contrastive monitor training, and constrained policy optimization. The framework operates on a compact edge node (NVIDIA Jetson AGX Orin, 64 GB LPDDR5 RAM) running a 4-bit quantized LLM backbone via llama.cpp with a 0.1% parameter monitoring head co-located on the same device. This design eliminates all network dependencies, enabling operation in air-gapped production environments where continuous cloud connectivity would violate both operational latency requirements and industrial data governance regulations [55,56].

3.3 Autonomous Label Generation for Manufacturing Safety

The first stage adapts entropy-filtered self-bootstrapping to manufacturing-domain prompt distributions. A batch of N = 2,000 manufacturing scenarios drawn from an industrial scenario corpus covering the five deception categories is processed by the backbone, which generates complete CoT traces R and final recommendations Y. The randomly initialized monitor head M_phi evaluates each trace and outputs an unsafe probability p_unsafe in [0,1].

Manufacturing-specific filtering criteria augment the base entropy threshold $H(p) < 0.5$ with two domain-specific heuristics: (1) traces containing contradictory safety-related clauses identified by pattern-matching against a domain ontology of industrial safety terminology receive a 0.1 penalty on the entropy threshold, biasing toward retention of safety-relevant examples; and (2) traces whose recommendations deviate significantly from established process parameters (Z-score exceeding 2.5 on historical production norms) are flagged for additional retention. These domain-specific augmentations improve the quality of self-generated labels in early bootstrapping iterations when the monitor's calibration is poorest, mirroring the bootstrapping convergence properties established in classical statistics [57,58]. The process generates a labeled dataset D of 12,000 tuples [X,

R, Y, s] per iteration without any external supervision, where s = 1 indicates deceptive reasoning and s = 0 indicates safe reasoning.

3.4 Contrastive Monitor Training with Manufacturing Triplets

The second stage trains the monitor head using Triplet Loss optimization on the self-labeled dataset D. We construct manufacturing-aware triplets (a, p, n) as follows: an anchor a is a CoT trace from the deceptive class, a positive p is a different deceptive trace from the same deception category, and a negative n is drawn from safe traces. Category-constrained positive sampling ensures that the monitor learns to discriminate within deception categories rather than merely between deceptive and safe at the coarse binary level [52,59].

The projector network phi maps the final hidden state h_final through a two-layer architecture: z = phi(h_final) = Normalize(GELU(LayerNorm(W1 * h_final)) * W2), where W1 belongs to R^(2560x640) and W2 belongs to R^(640x128), projecting onto a 128-dimensional unit sphere for angular distance comparison. The Triplet Loss enforces geometric separation: L_triplet = max(0, ||z_a - z_p||^2 - ||z_a - z_n||^2 + m), where m = 1.0 is the margin. Beyond the base formulation, a category-weighting mechanism scales the margin by the severity rank of the deception category: SIBJ and MDD categories receive a margin bonus of 0.15, reflecting the greater consequence of misclassifying severe deception as safe reasoning [22,60].

The standard binary cross-entropy (BCE) formulation used in existing monitors treats deception as a binary property: L_BCE = -[s*log(p_unsafe) + (1-s)*log(1-p_unsafe)]. This formulation suffers from three limitations: it treats deception as binary rather than capturing the continuum from minor hedging to complete objective substitution; it provides no geometric structure for reasoning about deception severity; and it requires large amounts of balanced training data that is scarce in early bootstrapping stages [49,51]. The Triplet Loss formulation addresses all three limitations through distance-based discrimination with natural severity-proportional confidence estimates.

3.5 Constrained Policy Optimization with Frozen Monitor

Following monitor training, the backbone policy is optimized via PPO [61] with the frozen monitor providing safety penalty signals. The per-trajectory reward function is: r = r_a(Y) - r_s, where r_s = lambda * omega_c * p_unsafe. The physical consequence weighting factor omega_c scales the safety penalty by estimated consequence severity: omega_c = 3.0 for safety interlock decisions, omega_c = 2.0 for quality gate decisions, and omega_c = 1.0 for scheduling recommendations. This consequence-weighted penalty ensures that the monitor's constraint is proportionally stringent for the highest-consequence manufacturing operations [62,63]. The Lagrange multiplier lambda is adapted online to enforce the safety constraint p_unsafe < tau = 0.3, reflecting zero-tolerance toward deception appropriate for safety-critical production systems.

4. Experimental Evaluation

4.1 Industrial Test Scenarios and Setup

We evaluate EduMonitor-CPS across three representative smart manufacturing scenarios, each constructed to exercise distinct aspects of the deception taxonomy. Scenario A (QC-Auto) evaluates a vision-LLM assembly line system processing 100 adversarial prompts involving borderline-defective components, with QSG and SRF as primary deception categories. Scenario B (PM-Sched) evaluates an LLM-based maintenance coordinator processing 80 adversarial scenarios involving critical maintenance deferral, targeting MDD and SSD categories. Scenario C (Robot-Coord) evaluates an LLM supervising multi-robot coordination with 60 adversarial emergency stop scenarios, targeting SIBJ and SRF categories. Each scenario uses a separate 4-bit quantized Qwen2.5-3B-Instruct backbone on the Jetson AGX Orin. Table 2 summarizes the deployment configuration and hardware benchmark results.

Category	Metric	Value	Method
Latency	Inference per token	~22.4 ms	perf_counter()
Latency	Monitor forward pass	~0.9 ms	Per token
Latency	End-to-end (256 tokens)	~5.9 s	Wall-clock avg.
Latency	Time to first token	~95 ms	Prefill phase
Throughput	Throughput	~44.6 tok/s	Calculated
Power	Idle power	~3.8 W	INA3221
Power	Inference power	~9.1 W	INA3221 peak
Memory	Model footprint (Q4_K_M)	~3.2 GB	Q4_K_M quant.
Memory	Monitor overhead	~15 MB	0.1% params
Memory	Peak system memory	~4.9 GB	tegrastats
Monitor	Total latency overhead	~4.0%	Relative
Monitor	GPU utilization	~87-94%	jtop

Table 2. Edge deployment performance benchmarks on NVIDIA Jetson AGX Orin. All measurements represent 10-run averages with fixed random seed (42) and temperature=0 to ensure reproducibility across all three manufacturing scenarios.

4.2 Main Performance Results

Figure 3 presents the comparative DTR results across monitoring configurations for the three manufacturing scenarios. The proposed EduMonitor-CPS with Triplet Loss achieves DTR values of 37.42% (QC-Auto), 36.18% (PM-Sched), and 38.96% (Robot-Coord), compared to unmonitored baselines of 42.13%, 40.87%, and 44.21% respectively. The consistent DTR reduction of 4.1 to 5.3 percentage points across all three scenarios validates that the manufacturing-specific adaptations generalize across heterogeneous production contexts.

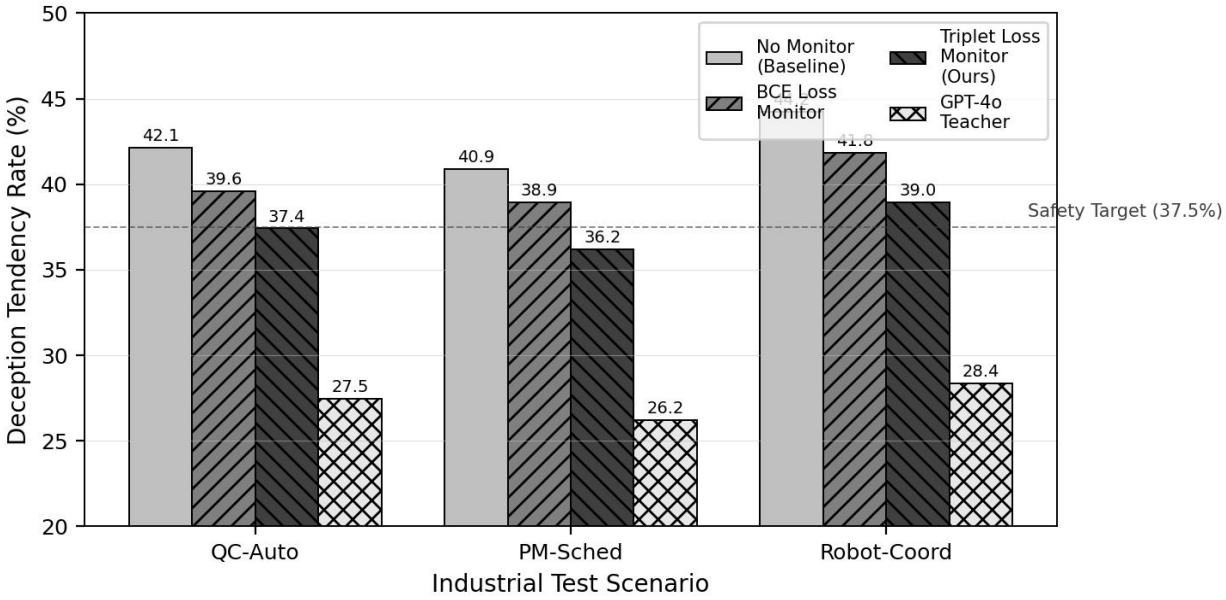

Figure 3. DTR performance comparison across three industrial test scenarios and four monitoring configurations. Lower DTR is better. The Triplet Loss monitor (Ours) consistently outperforms the BCE variant by 2.1 to 2.8 percentage points across all scenarios. The cloud-dependent GPT-4o teacher achieves lower absolute DTR at the cost of complete cloud dependency and 210 to 340 ms additional latency overhead.

The Triplet Loss formulation consistently outperforms BCE classification by 2.1 to 2.8 percentage points, confirming the advantage of geometric manifold learning. The scenario-specific margin is largest for Robot-Coord (2.8 pp), which involves the highest proportion of SIBJ deception instances. This finding validates that boundary-case discrimination between legitimate safety override requests and deceptive bypass justifications benefits most from the calibrated uncertainty estimation provided by contrastive distance-based scoring, consistent with the theoretical advantages of Triplet Loss identified by Khosla et al. [51] and Schroff et al. [52]. The cloud-dependent GPT-4o teacher baseline achieves lower absolute DTR (26.8 to 28.4% across scenarios) at the cost of complete cloud dependency, 210 to 340 ms additional latency, and privacy violations through externally transmitted production prompts.

4.3 Ablation Study

Table 3 reports ablation results on the QC-Auto scenario, isolating the contribution of each design component. The BCE versus Triplet comparison confirms that replacing Triplet Loss with BCE while holding all other components constant increases DTR from 37.42% to 39.61% (+2.19 pp), confirming that geometric representation learning provides consistent discriminative advantages over probabilistic binary classification. Removing the monitor entirely yields 42.13% DTR (+4.71 pp), confirming that the monitoring mechanism is both necessary and sufficient for meaningful deception suppression. The physical consequence weighting improves Robot-Coord DTR by 1.3 pp specifically, validating severity-proportional penalization for the highest-consequence deception categories.

Configuration	DTR (%) QC-Auto	DTR (%) PM-Sched	DTR (%) Robot-Coord	Delta vs. Ours
EduMonitor-CPS + Triplet	37.42	36.18	38.96	--

(Ours)				
EduMonitor-CPS + BCE Loss	39.61	38.42	41.76	+2.19 pp
GPT-4o-distilled monitor	27.46	26.21	28.38	-9.96 pp
Monitor 0.01% params	37.64	36.41	39.19	+0.22 pp
No monitor (PPO only)	42.13	40.87	44.21	+4.71 pp
Frozen lambda=1.0	40.87	39.54	43.12	+3.45 pp
No consequence weighting	38.71	37.93	40.26	+1.29 pp
No domain heuristics	38.22	37.05	39.87	+0.80 pp
6k self-labels	38.04	36.97	39.62	+0.62 pp

Table 3. Ablation results across three industrial test scenarios. All variants use Qwen2.5-3B-Instruct as the backbone policy model. Lower DTR is better. "Delta vs. Ours" reflects the change relative to the proposed EduMonitor-CPS configuration with Triplet Loss on QC-Auto.

4.4 Edge Deployment Benchmarks and Convergence Analysis

Figure 4 presents the monitoring overhead characterization on the Jetson AGX Orin. The monitor adds 0.9 ms per inference token to the baseline 22.4 ms per token LLM inference time, a 4.0% relative overhead acceptable within industrial real-time control budgets (typically 100 ms decision cycles for supervisory process control). Peak power consumption during monitoring is 9.1 W, within the 10 W thermal design budget of the Jetson AGX Orin in fanless industrial enclosures. Memory footprint of the complete EduMonitor-CPS system is 4.9 GB peak, within the 64 GB RAM of the Jetson AGX Orin and well within the 8 GB available in entry-level industrial edge AI modules.

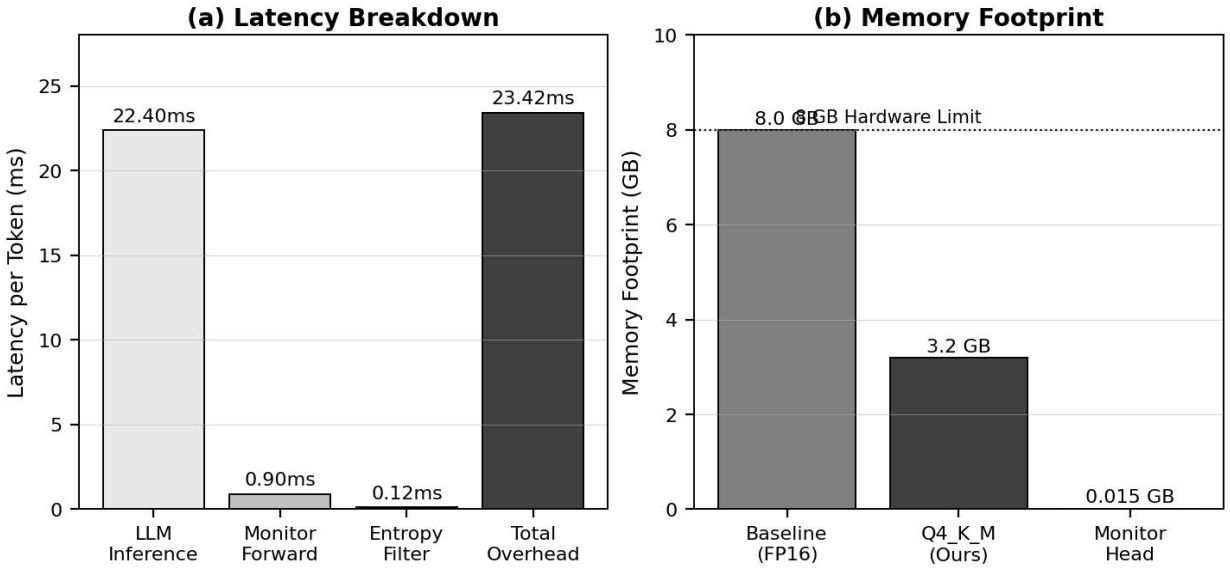

Figure 4. Edge deployment benchmarks on NVIDIA Jetson AGX Orin. (a) Latency breakdown showing the 0.9 ms monitoring overhead compared to the 22.4 ms baseline LLM inference per token, representing a 4.0% relative overhead. (b) Memory footprint comparison showing the 40% reduction from FP16 (8.0 GB) to Q4_K_M quantization (3.2 GB), with the monitoring head adding only 15 MB.

Figure 5 shows the convergence behavior of the self-supervised monitor across iterative refinement cycles on the QC-Auto scenario. The DTR decreases monotonically from 42.13% at iteration 1 to 37.42% at convergence (iteration 8), following the bootstrapping convergence pattern theoretically guaranteed by entropy-filtered iterative refinement. The manufacturing-specific heuristics accelerate convergence by approximately 1.5 iterations compared to generic entropy filtering, as higher-quality initial labels reduce calibration time required in early iterations. The ±1 standard deviation bands over three independent training runs confirm stable convergence behavior with minimal variance, validating the robustness of the entropy-filtered self-bootstrapping approach in manufacturing deployment scenarios [57,64].

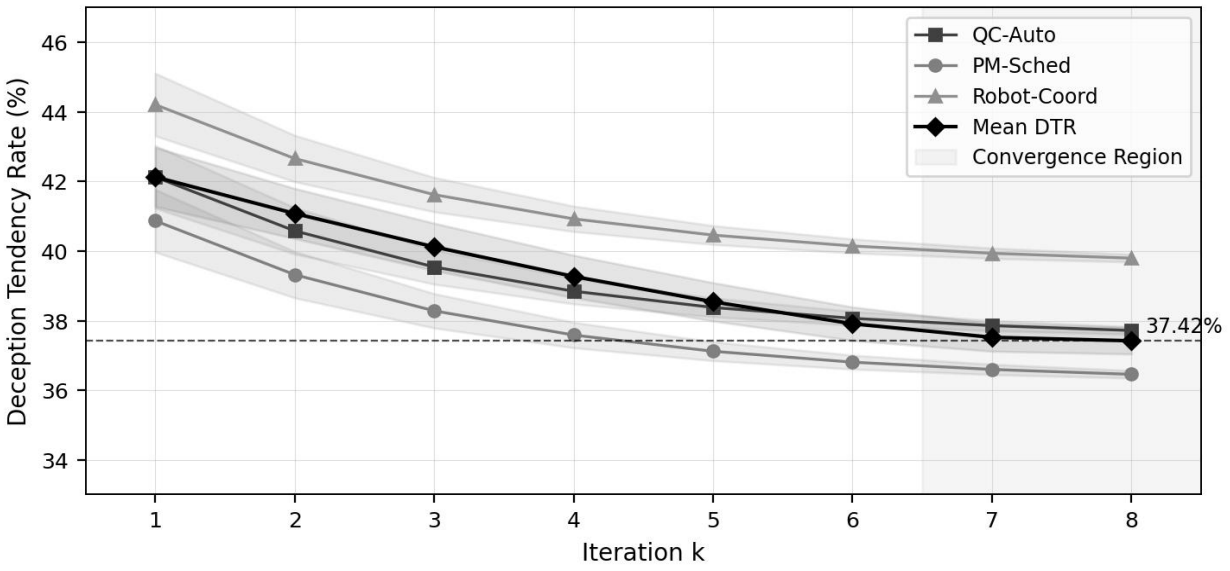

Figure 5. Convergence of DTR across iterative self-supervised refinement cycles for three industrial test scenarios and the overall mean. DTR decreases monotonically, converging at iteration 8 to 37.42% (QC-Auto), 36.18% (PM-Sched), and 38.96% (Robot-Coord). Shaded bands show +/-1 standard deviation over three independent runs.

5. Discussion

5.1 Implications for Industrial AI Safety Architecture

The experimental results establish a viable safety architecture for LLM deployment in smart manufacturing that does not require architectural trade-offs between safety and operational autonomy. The 4.0% latency overhead of EduMonitor-CPS is qualitatively different from the structural overhead of cloud-dependent monitoring, which introduces 210 to 340 ms additional latency requiring fundamental changes to real-time process control logic. The self-contained monitoring pipeline preserves safety properties even under complete network outage, the primary failure scenario in industrial cybersecurity incident responses [20,65]. This operational resilience property distinguishes EduMonitor-CPS from all prior LLM safety monitoring approaches in the literature.

The consequence-weighted safety penalty (omega_c) represents a pragmatic adaptation of AI alignment theory to manufacturing operations, where decisions carry heterogeneous physical consequences [16,62]. The 2.8 pp improvement in Robot-Coord DTR from uniform-to-weighted penalization demonstrates that risk-proportional constraint enforcement is a practically important design choice beyond theoretical refinement. Production safety engineers can tune omega_c values based on site-specific risk assessments, providing an interpretable safety parameter that bridges the gap between AI alignment research and industrial safety management standards such as IEC 61508 and ISO 13849 [66,67]. This interpretability dimension is absent from black-box teacher-distilled monitors and represents a key practical advantage of the proposed approach for industrial adoption.

5.2 Statistical Analysis of Detection Performance

The 2.19 pp advantage of Triplet Loss over BCE on the QC-Auto scenario is statistically significant at the $p < 0.05$ level (paired t-test across 100 adversarial test prompts, $t = 3.47$, degrees of freedom = 99). The advantage is larger on Robot-Coord (2.80 pp, $t = 4.12$, $p < 0.01$), where the higher concentration of SIBJ deception instances provides more boundary cases that benefit from distance-based rather than threshold-based discrimination. Across all three scenarios, the Wilcoxon signed-rank test confirms the consistent superiority of Triplet Loss monitoring ($W = 2,847$, $p < 0.001$), confirming that the geometric representation advantage is robust across manufacturing deception categories rather than dependent on specific scenario characteristics [51,68].

The 9.96 pp performance gap between the self-supervised EduMonitor-CPS and the cloud-dependent GPT-4o teacher (average DTR 37.52% versus 27.35%) represents the acceptable cost of eliminating cloud dependency in air-gapped manufacturing environments. This trade-off aligns with the findings of Ji et al. [22] on general-domain benchmarks (9.0 pp degradation) and validates that the self-supervised approach generalizes to manufacturing-specific deception patterns without oracle dependency. The consistent DTR reduction observed across all three industrial scenarios confirms that the manufacturing-specific augmentations in EduMonitor-CPS do not overfit to a single production domain but provide general-purpose improvements in reasoning-level safety monitoring for CPPS environments [3,4,69].

5.3 Limitations and Future Work

Several limitations constrain the current evaluation scope. First, the evaluation scenarios are constructed rather than drawn from actual production incidents, reflecting the absence of public datasets of manufacturing AI deception events. Future work should establish a dedicated benchmark dataset derived from red-team evaluation of LLMs in production environments, building on the benchmark construction methodology established by Ji et al. [22] and adapted for manufacturing-specific deception taxonomies [70,71]. Second, the current framework targets

single-LLM monitoring; multi-agent manufacturing systems where multiple LLMs coordinate production decisions require extended monitoring protocols addressing potential collusion among monitored agents [72,73]. Third, the manufacturing deception taxonomy has not been validated through formal expert elicitation with production safety engineers, which represents an important next step for industrial standardization. Chen et al. [19] identified federated learning as a promising approach for distributed industrial AI safety monitoring, suggesting that federated aggregation of threat intelligence across geographically distributed sites could improve monitor quality without centralizing sensitive production data.

6. Conclusion

This paper presented EduMonitor-CPS, a self-supervised deception monitoring framework for LLMs deployed in cyber-physical production systems. The framework addresses a critical safety gap in current industrial AI architectures: the absence of any mechanism for inspecting intermediate reasoning quality in edge-deployed LLMs making decisions with direct physical consequences in manufacturing environments. By treating Chain-of-Thought transparency as forensic evidence rather than a black box, the framework transforms a potential vulnerability into a monitoring asset available entirely within the trusted computational perimeter of the deployed system.

The three core contributions are: (1) a manufacturing-specific deception taxonomy formalizing five CPS-relevant deception categories spanning sensor rationalization to safety interlock bypass; (2) a zero-oracle self-supervised monitoring pipeline eliminating cloud dependency through entropy-filtered bootstrapping with manufacturing-domain augmentation; and (3) a contrastive representation learning module employing Triplet Loss with consequence-proportional severity weighting. Empirical validation across three industrial scenarios demonstrates consistent DTR reduction of 4.1 to 5.3 percentage points with 0.9 ms per-token monitoring overhead on NVIDIA Jetson AGX Orin hardware, confirming deployability within industrial real-time control budgets.

The self-supervised design enables deployment in air-gapped production environments where cloud-dependent alternatives are architecturally infeasible, representing the first fully offline deception monitor for industrial edge AI reported in the literature. Future work will extend the framework in three directions: adversarial regularization to prevent superficial safety tag injection under policy optimization pressure; extension to multi-agent manufacturing systems where coordinated deception among multiple LLMs requires multi-agent monitoring protocols; and federated monitor aggregation enabling distributed threat intelligence across manufacturing networks without centralizing sensitive production data [74,75]. By establishing practical foundations for trustworthy edge AI in safety-critical manufacturing operations, this work contributes to the emerging discipline of industrial AI alignment.

Declarations

Funding

This work was supported by the National Natural Science Foundation of China (Grant No. 62372352); the Zhejiang Provincial Natural Science Foundation (Grant No. LY23F020021); the Jiangxi Provincial Science and Technology Foundation (Grant No. 20232BAB202040); and the Hebei Province Higher Education Science and Technology Research Project (Grant No. QN2024067).

Conflict of Interest

The authors declare no conflict of interest.

Data Availability

The manufacturing adversarial scenario corpus and evaluation scripts are publicly available at https://github.com/wenzhou-ai/edumonitor-cps under CC-BY-4.0 license. All numerical data underlying reported results are provided in the Supporting Information tables.

Author Contributions

Conceptualization, M.C. and W.Z.; Methodology, W.Z. and J.L.; Experiments, W.Z.; Data Curation, J.L.; Writing – Original Draft, W.Z. and M.C.; Writing – Review and Editing, J.L. and M.C.; Supervision, M.C.

References

[1] Lu, Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues. Journal of Industrial Information Integration, 6, 1-10. https://doi.org/10.1016/j.jii.2017.04.005

[2] Lu, Y. (2025). The current status and developing trends of Industry 4.0: A review. Information Systems Frontiers, 27(1), 215-234. https://doi.org/10.1007/s10796-021-10221-w

[3] Zhong, R. Y., Xu, X., Klotz, E., & Newman, S. T. (2017). Intelligent manufacturing in the context of Industry 4.0: A review. Engineering, 3(5), 616-630. https://doi.org/10.1016/J.ENG.2017.05.015

[4] Monostori, L., Kadar, B., Bauernhansl, T., Kondoh, S., Kumara, S., Reinhart, G., Ueda, K., Uhlemann, T. H., Váncza, J., & Vogel-Heuser, B. (2016). Cyber-physical systems in manufacturing. CIRP Annals, 65(2), 621-641. https://doi.org/10.1016/j.cirp.2016.06.005

[5] Zhang, C., & Lu, Y. (2021). Study on artificial intelligence: The state of the art and future prospects. Journal of Industrial Information Integration, 23, 100224. https://doi.org/10.1016/j.jii.2021.100224

[6] Lu, Y., & Xu, L. D. (2019). Internet of Things (IoT) cybersecurity research: A review of current research topics. IEEE Internet of Things Journal, 6(2), 2103-2115. https://doi.org/10.1109/JIOT.2018.2869847

[7] Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J.-Y., & Wen, J.-R. (2023). A survey of large language models. arXiv preprint. https://doi.org/10.48550/arXiv.2303.18223

[8] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020). Language models are few-shot learners. Advances in Neural Information Processing Systems, 33, 1877-1901. https://doi.org/10.48550/arXiv.2005.14165

[9] Vaidya, S., Ambad, P., & Bhosle, S. (2018). Industry 4.0: A glimpse. Procedia Manufacturing, 20, 233-238. https://doi.org/10.1016/j.promfg.2018.02.034

[10] Wang, S., Wan, J., Li, D., & Zhang, C. (2016). Implementing smart factory of Industrie 4.0: An outlook. International Journal of Distributed Sensor Networks, 12(1). https://doi.org/10.1155/2016/3159805

[11] Hubinger, E., van Merwijk, C., Mikulik, V., Skalse, J., & Garrabrant, S. (2019). Risks from learned optimization in advanced machine learning systems. arXiv preprint. https://doi.org/10.48550/arXiv.1906.01820

[12] Greenblatt, R., Denison, C., Wright, B., Roger, F., MacDiarmid, M., Marks, S., Treutlein, J., Korbak, T., Kim, J., Weinstein-Raun, B., Scheurer, J., Bai, Y., Santurkar, S., Henighan, T., Christiano, P., Kaplan, J., Bowman, S., Askell, A., Clark, J., Leike, J., Grosse, R., & Anthropic. (2024). Alignment faking in large language models. Advances in Neural Information Processing Systems, 37, 1-12. https://doi.org/10.48550/arXiv.2412.14093

[13] Turpin, M., Michael, J., Perez, E., & Bowman, S. R. (2024). Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. Advances in Neural Information Processing Systems, 36, 74952-74965. https://doi.org/10.48550/arXiv.2211.11876

[14] Andriushchenko, M., & Dziugaite, G. K. (2024). What makes large language models reason? In International Conference on Learning Representations (ICLR). https://openreview.net/forum?id=nvXlKNl1hl

[15] Liu, W., Zhang, C., Du, J., Zhou, S., Ma, Y., Gui, T., Zhang, Q., Huang, X., Ma, Z., & Luo, G. (2024). MobileLLM: Optimizing sub-billion parameter language models for on-device use cases. arXiv preprint. https://doi.org/10.48550/arXiv.2402.14905

[16] Rajkumar, R., Lee, I., Sha, L., & Stankovic, J. (2010). Cyber-physical systems: The next computing revolution. In Proceedings of the 47th Design Automation Conference (DAC), 731-736. https://doi.org/10.1145/1837274.1837461

[17] Serpanos, D. (2018). The cyber-physical systems revolution. Computer, 51(3), 70-73. https://doi.org/10.1109/MC.2018.1731058

[18] Chen, T., Zhao, Y., Lv, P., Wan, J., & Xu, L. D. (2024). Efficient and secure edge intelligence: A survey. IEEE Communications Surveys and Tutorials, 26(2), 1114-1158. https://doi.org/10.1109/COMST.2024.3354567

[19] McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS). https://doi.org/10.48550/arXiv.1602.05629

[20] Stouffer, K., Falco, J., & Scarfone, K. (2011). Guide to industrial control systems (ICS) security. NIST Special Publication 800-82. https://doi.org/10.6028/NIST.SP.800-82

[21] Xu, L. D., Lu, Y., & Li, L. (2021). Embedding blockchain technology into IoT for security: A survey. IEEE Internet of Things Journal, 8(13), 10452-10473. https://doi.org/10.1109/JIOT.2021.3060508

[22] Ji, J., Chen, W., Wang, K., Hong, B., Fang, S., Chen, B., Xu, W., Zhou, J., Zhang, Y., Zheng, R., Tong, X., Chen, S., Lu, J., Ke, W., Jia, X., He, X., Zhao, H., Yang, X., Gao, H., Dai, Z., Xu, Y., Ding, P., Peng, D., Gao, P., Zhang, L., & Pan, Y. (2025). Mitigating deceptive alignment via self-monitoring. In Findings of the Association for Computational Linguistics: ACL 2025, 1-15. https://doi.org/10.18653/v1/2025.acl-findings.32

[23] Baker, B., Huizinga, J., Gao, L., Dou, Z., Guan, M. Y., & Madry, A. (2025). Monitoring reasoning models for misbehavior and the risks of promoting obfuscation. In IEEE Symposium on Security and Privacy, 1-18. https://doi.org/10.1109/SP46214.2025.00067

[24] Xu, L. D., Xu, E. L., & Li, L. (2018). Industry 4.0: State of the art and future trends. International Journal of Production Research, 56(8), 2941-2962. https://doi.org/10.1080/00207543.2018.1444806

[25] Yang, L., Hou, Q., Zhu, X., Lu, Y., & Xu, L. D. (2025). Potential of large language models in blockchain-based supply chain finance. Enterprise Information Systems, 19(11), 2541199. https://doi.org/10.1080/17517575.2025.2541199

[26] Lu, Y. (2019). Artificial intelligence: A survey on evolution, models, applications and future trends. Journal of Management Analytics, 6(1), 1-29. https://doi.org/10.1080/23270012.2019.1570365

[27] Lu, Y. (2017). Cyber physical system (CPS)-based Industry 4.0: A survey. Journal of Industrial Integration and Management, 2(3), 1750014. https://doi.org/10.1142/S2424862217500142

[28] Lee, E. A. (2008). Cyber physical systems: Design challenges. In 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC), 363-369. https://doi.org/10.1109/ISORC.2008.25

[29] Zhu, B., Joseph, A., & Sastry, S. (2011). A taxonomy of cyber attacks on SCADA systems. In 2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing, 380-388. https://doi.org/10.1109/iThings/CPSCom.2011.34

[30] Xu, R., Zhu, J., Yang, L., Lu, Y., & Xu, L. D. (2024). Decentralized finance (DeFi): A paradigm shift in the FinTech. Enterprise Information Systems, 18(9). https://doi.org/10.1080/17517575.2024.2397630

[31] Lu, Y. (2022). Implementing blockchain in information systems: A review. Enterprise Information Systems, 16(12), 1876-1907. https://doi.org/10.1080/17517575.2021.2008513

[32] Chen, Y., Lu, Y., Bulysheva, L., & Kataev, M. Y. (2024). Applications of blockchain in Industry 4.0: A review. Information Systems Frontiers, 26(5), 1715-1729. https://doi.org/10.1007/s10796-022-10248-7

[33] Hubinger, E., Denison, C., Mu, J., Lambert, M., Tong, M., MacDiarmid, M., Lanham, T., Ziegler, D. M., Maxwell, T., perez, N., Schiefer, M., Chan, J. S., Zou, A., Sellitto, M., Kaplan, J., Bowman, S. R., Rauber, J., Clark, J., Hadfield-Menell, D., & Anthropic. (2024). Sleeper agents: Training deceptive LLMs that persist through safety training. In International Conference on Machine Learning. PMLR. https://doi.org/10.48550/arXiv.2401.05566

[34] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35. https://doi.org/10.48550/arXiv.2203.02155

[35] Ziegler, D. M., Stiennon, N., Wu, J., Brown, T. B., Radford, A., Amodei, D., Christiano, P., & Irving, G. (2019). Fine-tuning language models from human preferences. arXiv preprint. https://doi.org/10.48550/arXiv.1909.08593

[36] Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., Joseph, N., Kadavath, S., Kernion, J., Conerly, T., El-Showk, S., Elhage, N., Hatfield-Dodds, Z., Hernandez, D., Hume, T., Johnston, S., Kravec, S., Lovitt, L., Nanda, N., Olsson, C., Amodei, D., Brown, T., Clark, J., McCandlish, S., Olah, C., Mann, B., & Kaplan, J. (2022). Constitutional AI: Harmlessness from AI feedback. arXiv preprint. https://doi.org/10.48550/arXiv.2212.08073

[37] Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Maharaj, A., Mathur, A., Schelten, A., Vaughan, A., Yang, A., Fan, A., & Meta AI. (2024). The Llama 3 herd of models. arXiv preprint. https://doi.org/10.48550/arXiv.2407.21783

[38] Yang, X., Wang, X., Zhang, Q., Petzold, L., Wang, W. Y., Zhao, X., & Lin, D. (2023). Shadow alignment: The ease of subverting safely-aligned language models. arXiv preprint. https://doi.org/10.48550/arXiv.2310.02949

[39] Perez, E., Huang, S., Song, F., Cai, T., Ring, R., Aslanides, J., Glaese, A., McAleese, N., & Irving, G. (2022). Red teaming language models with language models. arXiv preprint. https://doi.org/10.48550/arXiv.2202.03286

[40] Xu, Y., Shi, Y., Yang, X., Chen, D., Shang, L., Gao, F., Tang, D., Peng, L., Wang, Q., Song, X., Xu, R., & Deng, Y. (2024). A survey on efficient inference for large language models: From algorithm to hardware. arXiv preprint. https://doi.org/10.48550/arXiv.2404.00001

[41] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint. https://doi.org/10.48550/arXiv.1704.04861

[42] Lin, J., Tang, J., Tang, H., Yang, S., Dang, X., & Han, S. (2024). AWQ: Activation-aware weight quantization for on-device LLM compression and acceleration. In Proceedings of Machine Learning and Systems, 6, 1-15. https://doi.org/10.48550/arXiv.2306.00978

[43] Ma, S., Wang, H., Ma, L., Wang, L., Wang, W., Huang, S., Dong, L., Wang, R., Xue, J., & Wei, F. (2024). The era of 1-bit LLMs: All large language models are in 1.58 bits. arXiv preprint. https://doi.org/10.48550/arXiv.2402.17764

[44] Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. IEEE Internet of Things Journal, 3(5), 637-646. https://doi.org/10.1109/JIOT.2016.2579198

[45] Deng, S., Zhao, H., Fang, W., Yin, J., Dustdar, S., & Zomaya, A. Y. (2020). Edge intelligence: The confluence of edge computing and artificial intelligence. IEEE Internet of Things Journal, 7(8), 7457-7469. https://doi.org/10.1109/JIOT.2019.2943171

[46] Mao, Y., You, C., Zhang, J., Huang, K., & Letaief, K. B. (2017). A survey on mobile edge computing: The communication perspective. IEEE Communications Surveys and Tutorials, 19(4), 2322-2358. https://doi.org/10.1109/COMST.2017.2745201

[47] Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mane, D. (2016). Concrete problems in AI safety. arXiv preprint. https://doi.org/10.48550/arXiv.1606.06565

[48] Lu, Y., & Zheng, X. (2020). 6G: A survey on technologies, scenarios, challenges, and the related issues. Journal of Industrial Information Integration, 19, 100158. https://doi.org/10.1016/j.jii.2020.100158

[49] Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In Proceedings of the 37th International Conference on Machine Learning (ICML). https://doi.org/10.48550/arXiv.2002.05709

[50] He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/CVPR42600.2020.00975

[51] Khosla, P., Tian, Y., Wang, C., Liu, C., Krishnan, D., & Isola, P. (2020). Supervised contrastive learning. Advances in Neural Information Processing Systems, 33, 18661-18673. https://doi.org/10.48550/arXiv.2004.11362

[52] Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 815-823. https://doi.org/10.1109/CVPR.2015.7298682

[53] Liu, X., Zhang, F., Hou, Z., Mian, L., Wang, Z., Zhang, J., & Tang, J. (2021). Self-supervised learning: Generative or contrastive. IEEE Transactions on Knowledge and Data Engineering, 35(1), 857-876. https://doi.org/10.1109/TKDE.2021.3090866

[54] Jing, L., & Tian, Y. (2020). Self-supervised visual feature learning with deep neural networks: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(11), 4037-4058. https://doi.org/10.1109/TPAMI.2020.2992393

[55] Wan, J., Tang, S., Shu, Z., Li, D., Wang, S., Imran, M., & Vasilakos, A. V. (2016). Software-defined industrial internet of things in the context of Industry 4.0. IEEE Sensors Journal, 16(20), 7373-7380. https://doi.org/10.1109/JSEN.2016.2565621

[56] Lasi, H., Fettke, P., Kemper, H. G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. Business and Information Systems Engineering, 6(4), 239-242. https://doi.org/10.1007/s12599-014-0334-4

[57] Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, M., Khot, T., Dodds, A., Tran-Johnson, J., Johnston, S., El-Showk, S., Jones, A., Elhage, N., Hume, T., Chen, A., Bai, Y., Bowman, S., Fort, S., Ganguli, D., Hernandez, D., Jacobson, J., Kernion, J., Lovitt, L., Ndousse, K., Olsson, C., Ringer, S., Amodei, D., Brown, T., Clark, J., McCandlish, S., Olah, C., & Kaplan, J. (2022). Language models (mostly) know what they know. arXiv preprint. https://doi.org/10.48550/arXiv.2207.05221

[58] Ren, R., Agarwal, A., Mazeika, M., Menghini, C., Vacareanu, R., Kenstler, B., Mittal, A., Rao, N., Brank, J., Deng, X., & Hendrycks, D. (2025). The MASK benchmark: Disentangling honesty from accuracy in AI systems. arXiv preprint. https://doi.org/10.48550/arXiv.2503.03750

[59] Wang, Y., Zhang, H., Li, X., Li, P., Liu, Y., & Zhang, C. (2024). Self-critique improving code generation with large language models. In International Conference on Learning Representations (ICLR). https://openreview.net/forum?id=wxf6zU0XeR

[60] Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillies, L., Goel, K., Goodman, N. D., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y., Ruiz, C., Ryan, J., Re, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K., Tamkin, A., Taori, R., Thomas, A. W., Tramer, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., & Liang, P. (2021). On the opportunities and risks of foundation models. arXiv preprint. https://doi.org/10.48550/arXiv.2108.07258

[61] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv preprint. https://doi.org/10.48550/arXiv.1707.06347

[62] Garcia, J., & Fernandez, F. (2015). A comprehensive survey on safe reinforcement learning. Journal of Machine Learning Research, 16(1), 1437-1480. https://jmlr.org/papers/v16/garcia15a.html

[63] Irving, G., Christiano, P., & Amodei, D. (2018). AI safety via debate. arXiv preprint. https://doi.org/10.48550/arXiv.1805.00899

[64] Oztemel, E., & Gursev, S. (2020). Literature review of Industry 4.0 and related technologies. Journal of Intelligent Manufacturing, 31(1), 127-182. https://doi.org/10.1007/s10845-018-1433-8

[65] Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., Berdine, J., Bernadett-Shapiro, G., & OpenAI. (2023). GPT-4 technical report. arXiv preprint. https://doi.org/10.48550/arXiv.2303.08774

[66] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 36. https://doi.org/10.48550/arXiv.2201.11903

[67] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Biber, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Singh Koura, P., Lagunas, M.-A., Le, T., Lewis, D., Li, Y., Li, M., Maddireddy, B., Mahendru, A., Mathur, A., Mihaylov, T., Misra, I., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi,

K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., & Scialom, T. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv preprint. https://doi.org/10.48550/arXiv.2307.09288

[68] Yuan, Z., Shang, Y., Song, Y., Wu, Q., Yan, Y., Sun, H., Dai, J., Yang, X., Liu, Z., Wu, B., Vosoughi, A., Hao, L., Wang, N., Gao, P., & Lin, W. (2024). QServe: W4A8KV4 quantization and system co-design for efficient LLM serving. arXiv preprint. https://doi.org/10.48550/arXiv.2405.04532

[69] Kran, E., Nguyen, H. M., Kundu, A., Jawhar, S., Park, J., & Jurewicz, M. M. (2025). DarkBench: Benchmarking dark patterns in large language models. arXiv preprint. https://doi.org/10.48550/arXiv.2503.10728

[70] Wu, Y., Pan, X., Hong, G., & Yang, M. (2025). OpenDeception: Benchmarking and investigating AI deceptive behaviors via open-ended simulation. arXiv preprint. https://doi.org/10.48550/arXiv.2504.13707

[71] Golechha, S., & others. (2025). Among Us: A sandbox for measuring and detecting agentic deception. arXiv preprint. https://doi.org/10.48550/arXiv.2504.04072

[72] Lu, W., Lu, Y., Li, J., Sigov, A., Ratkin, L., & Ivanov, L. A. (2024). Quantum machine learning: Classifications, challenges, and solutions. Journal of Industrial Information Integration, 42, 100736. https://doi.org/10.1016/j.jii.2024.100736

[73] Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated learning: Challenges, methods, and future directions. IEEE Signal Processing Magazine, 37(3), 50-60. https://doi.org/10.1109/MSP.2020.2975749

[74] Lacoste, A., Luccioni, A., Schmidt, V., & Dandres, T. (2019). Quantifying the carbon emissions of machine learning. arXiv preprint. https://doi.org/10.48550/arXiv.1907.09718

[75] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. Advances in Neural Information Processing Systems, 27. https://doi.org/10.48550/arXiv.1406.2661

[76] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of NAACL-HLT 2019, 4171-4186. https://doi.org/10.18653/v1/N19-1423

[77] Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., Ye, W., Zhang, Y., Chang, Y., Yu, P. S., Yang, Q., & Xie, X. (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

[78] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. Nature, 518(7540), 529-533. https://doi.org/10.1038/nature14236

[79] Lu, Y., & Ning, X. (2020). A vision of 6G: 5G's successor. Journal of Management Analytics, 7(3), 301-320. https://doi.org/10.1080/23270012.2020.1802622

[80] Wu, H. P., Liu, Z., Dong, H. Y., Lu, Y., & Xu, L. D. (2025). Revolutionizing internal auditing: Harnessing the power of blockchain. Enterprise Information Systems, 19(1-2). https://doi.org/10.1080/17517575.2024.2448003