International Conference on Science, Engineering & Management Trends

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

Agentic AI and Foundation Model Systems: Architectures, Multimodal Intelligence, and Latency Optimization

Atharva Pisal, Soham Pawar, Revan Chenna, Swaraj Dalvi
Department of Electronics and Computer Science Shah & Anchor Kutchhi Engineering College, India

Abstract- Agentic AI represents a paradigm shift in which intelligent systems autonomously reason, plan, and act within dynamic environments to achieve complex goals with minimal human intervention. This paper presents a comprehensive study of agentic AI architectures integrating large-scale foundation models and multimodal Visual Language Models (VLMs). Key architectural components including the LLM reasoning core, memory modules, planning engines, tool interaction layers, and multi-agent orchestration mechanisms are analyzed in depth. Latency challenges arising from model inference, framework processing, communication overhead, and system inefficiencies are systematically examined, and optimization strategies including model compression, speculative decoding, KV-cache management, and edge deployment are evaluated. Applications spanning indus-trial automation, healthcare, energy systems, smart infrastructure, and financial services demonstrate significant performance im-provements. Open challenges relating to computational complexity, hallucination, interpretability, privacy, and ethical alignment are discussed, followed by future research directions toward efficient, trustworthy, and scalable agentic AI systems.

Keywords- Agentic AI, Foundation Models, Visual Language Models, Multi-Agent Systems, Latency Optimization, Transformer Architectures, Multimodal Learning, Autonomous Systems

I. INTRODUCTION

Artificial Intelligence has evolved significantly from rule-based symbolic systems to highly adaptive, data-driven ar-chitectures. Early AI approaches relied on handcrafted rules and expert knowledge, which limited scalability and hindered generalization to unseen scenarios. The introduction of deep learning, and subsequently transformer-based architectures, enabled systems to learn hierarchical representations from large-scale data, achieving human-competitive performance across perception, language, and reasoning benchmarks.

Foundation models have fundamentally changed the AI landscape. Trained on massive corpora through self-supervised objectives, these models demonstrate strong generalization and can be adapted to diverse downstream tasks with minimal fine-tuning. Their scale-driven emergent capabilities, including few-shot learning and chain-of-thought reasoning, represent a qualitative leap beyond earlier narrow AI systems [7].

Agentic AI further extends this paradigm by introducing goal-directed autonomy. Rather than passively responding to queries, agentic systems decompose high-level objectives, formulate multi-step plans,

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

interact with external tools and environments, and refine their behavior through feedback loops. This transition from reactive inference to proactive, closed-loop operation marks a critical advancement toward general-purpose intelligent systems [1].

Multimodal intelligence has become equally essential for real-world deployments. Visual Language Models (VLMs) unify visual perception and linguistic reasoning within a single architecture, enabling context-aware decision-making in tasks ranging from medical image analysis to industrial inspection [2]. The convergence of agentic behavior, foundation model reasoning, and multimodal perception introduces new challenges in system design, latency management, and reliable deployment [3].

Key contributions of this paper:

- A detailed architectural analysis of agentic AI systems covering all core components and their interactions
- A formal latency decomposition model with quantified optimization strategies
- A comprehensive review of real-world applications with performance benchmarks
- A structured discussion of open challenges and future research directions

The remainder of this paper is organized as follows. Sec-tion II reviews background and related work. Section III details agentic AI architecture. Section IV analyzes latency challenges and optimizations. Section V presents applications. Sections VI and VII discuss challenges and future directions before concluding in Section VIII.

II. BACKGROUND AND RELATED WORK

A. Evolution of Artificial Intelligence

AI research has progressed through distinct phases, each defined by a dominant paradigm. Symbolic AI encoded domain

TABLE I

Model Type	Data	Accuracy	Flexibility
CNN / RNN	Unimodal	70%	Low
Large Language Model	Text	85%	Medium
Foundation Model	Multi-task	90%	High
Multimodal FM	Vision+Text	93%	Very High

knowledge through explicit logical rules, achieving strong performance in constrained, well-defined problems while failing to generalize under uncertainty or incomplete information.

Statistical machine learning shifted the focus to data-driven model induction. Methods such as support vector machines, gra-dient boosting, and Bayesian networks demonstrated practical utility across classification, regression, and anomaly detection tasks, though they required significant feature engineering for unstructured data.

Deep learning resolved the feature engineering bottleneck through hierarchical representation learning. Convolutional Neural Networks (CNNs) achieved state-of-the-art performance in image recognition, while Long Short-Term Memory (LSTM) networks advanced sequence modeling for speech and text. The introduction of the attention mechanism and the transformer architecture [8] marked a turning point, enabling parallel processing of long sequences with superior modeling of global dependencies. Scaling laws subsequently demonstrated consistent capability improvements with increased model parameters, data, and compute [7].

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

B. Foundation Models

Foundation models are large-scale models pretrained on diverse, broad datasets and adapted to downstream tasks through fine-tuning or in-context learning. Their defining characteristics include:

- Large-scale pretraining: Trained on corpora spanning billions to trillions of tokens
- Transfer learning: Pretraining knowledge transfers effec-tively to new domains
- Emergent capabilities: Complex behaviors such as multi-step reasoning emerge at scale
- Few-shot generalization: Tasks are learned from exam-ples in the prompt context
- Domain adaptability: Fine-tuning techniques such as LoRA enable cost-effective specialization

Parameter-efficient fine-tuning via Low-Rank Adaptation (LoRA) and Retrieval-Augmented Generation (RAG) [10] extend foundation models to domain-specific applications with minimal additional training. RAG architectures ground model outputs in external knowledge bases, significantly reducing hallucination and improving factual accuracy in knowledge-intensive tasks.

Table I summarizes the progressive improvement in accuracy and flexibility across AI model paradigms. Foundation models substantially reduce the need for task-specific architectures and enable scalable solutions across industries.

C. Visual Language Models

Visual Language Models (VLMs) integrate visual perception and language understanding within a unified architecture, enabling multimodal reasoning over images and text simulta-neously [2]. A typical VLM architecture comprises:

- **Vision Encoder:** Extracts spatial and semantic features using Vision Transformers (ViT) or convolutional back-bones
- **Language Encoder:** Processes textual queries and instruc-tions using transformer-based LLMs
- **Alignment Module:** Projects visual features into the language model's embedding space via cross-attention or linear projection layers
- **Fusion Layer:** Combines visual and textual representa-tions for joint reasoning and generation

Prominent VLMs including CLIP [12], BLIP-2, LLaVA, and Flamingo have demonstrated strong performance on visual question answering, image captioning, and visual grounding benchmarks. CLIP introduced zero-shot visual classification through contrastive image-text pretraining. LLaVA and BLIP-2 further bridged frozen visual encoders with instruction-tuned LLMs, enabling open-ended visual dialogue.

D. Multi-Agent Systems

Multi-agent systems extend the agentic paradigm to set-tings where multiple specialized agents collaborate toward individual or collective goals. Coordination mechanisms include centralized planning, decentralized peer-to-peer negotiation, and hierarchical supervisor-worker architectures. Frameworks such as AutoGen, CrewAI, and LangGraph have demonstrated that LLM-based multi-agent pipelines outperform single-agent approaches on complex reasoning, workflow automation, and software engineering tasks [1].

III. AGENTIC AI ARCHITECTURE

Agentic AI systems are composed of tightly integrated, modular components that collectively enable autonomous perception, reasoning, planning, and action. Unlike monolithic inference models, agentic architectures distribute cognition across specialized subsystems, each contributing a distinct capability to the overall intelligent behavior.

A. Large Language Model Core

The LLM serves as the central reasoning engine, responsible for interpreting inputs, generating action plans, coordinating component interactions, and synthesizing outputs. Built on the transformer

architecture [8], the LLM core uses self-attention to model contextual relationships across the full input sequence, supporting complex multi-step reasoning.

Within the agentic system, the LLM core performs instruc-tion parsing, goal decomposition, intermediate reasoning, tool selection, and output synthesis. Advanced prompting strategies including chain-of-thought reasoning [9], ReAct [11], and reflection mechanisms significantly improve accuracy on multi-step tasks by encouraging structured intermediate reasoning and iterative self-evaluation.

The trade-off between model capability and inference latency is a central design consideration. Larger models provide superior reasoning at greater computational cost, motivating the use of compression and acceleration techniques discussed in Section IV.

B. Memory Module

Persistent, context-aware behavior requires a structured memory system operating across multiple timescales:

- Working memory: The active context window main-taining the current task state, recent interactions, and intermediate reasoning steps
- Episodic memory: Vector database storage of past interactions, retrieved by semantic similarity using systems such as FAISS or Pinecone
- Semantic memory: Structured knowledge about facts, concepts, and domain relationships, stored in knowledge graphs or structured databases
- Procedural memory: Cached action sequences, reason-ing templates, and learned behavioral patterns enabling efficient task execution

RAG architectures [10] integrate episodic and semantic memory with LLM inference, enabling access to relevant historical context and domain knowledge without embedding it in model weights. Efficient indexing and retrieval are critical for minimizing memory-related latency contributions in the overall system.

C. Planning Engine

The planning module decomposes complex goals into executable action sequences through hierarchical task analysis, constraint evaluation, and resource-aware scheduling. Core planning functions include:

- Goal decomposition: Translating high-level objectives into ordered subtasks
- Dependency resolution: Identifying task ordering con-straints and parallelization opportunities
- Resource allocation: Assigning tools, APIs, and compute resources to planned steps
- Dynamic replanning: Adjusting plans in response to unexpected observations or tool failures

ReAct [11] and tree-of-thought planning explore multiple reasoning branches in parallel, selecting the most promising execution path based on heuristic or learned evaluation criteria. This adaptive planning enables robust task completion even under uncertainty and partial observability.

D. Tool Interaction Layer

The tool interaction layer connects the agent to external systems, extending its capabilities beyond internal language processing to real-world actions with measurable consequences. Integrated tool categories include:

TABLE II
ARCHITECTURAL COMPONENT ANALYSIS

Component	Function	Impact
LLM Core	Reasoning & generation	High
Memory Module	Context persistence	High
Planning Engine	Task decomposition	High
Tool Layer	External execution	Medium

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

Orchestrator	Multi-agent coordination	Medium
Feedback Loop	Adaptation & learning	Medium

- **Search and retrieval:** Web search APIs, document stores, and knowledge base interfaces
- **Code execution:** Python interpreters, Jupyter environ-ments, and sandboxed compute
- **Data systems:** SQL and NoSQL databases, analytics platforms, and visualization libraries
- **Communication APIs:** Email, calendar, and messaging system integrations
- **Domain-specific tools:** Medical databases, engineering simulators, and financial data platforms

Emerging standards including the Model Context Protocol (MCP) standardize tool interfaces, enabling agents to interact with a growing ecosystem of compatible services without bespoke integration engineering.

E. Orchestration and Agent Coordination

In multi-agent deployments, an orchestration layer coordi-nates individual agent activities, maintains global task state, resolves inter-agent dependencies, and aggregates distributed results. Orchestration strategies include sequential pipelines, parallel fan-out execution, debate-and-consensus mechanisms, and hierarchical supervisor-worker delegation.

System Workflow:

The agentic system operates in a continuous closed loop:
Perception → Reasoning → Planning → Execution → Feedback → Adaptation

IV. LATENCY CHALLENGES IN AI SYSTEMS

Latency is among the most critical performance bottle-necks in production agentic AI deployments. High latency degrades user experience, limits real-time applicability, and restricts adoption in safety-critical domains including robotics, autonomous vehicles, clinical decision support, and financial trading. Systematic analysis of latency sources is essential for targeted optimization.

A. Latency Decomposition Model

The total system latency can be formally decomposed as:

$$L_{total} = L_{model} + L_{framework} + L_{comm} + L_{sys} \quad (1)$$

where Lmodel is inference latency from neural computation, Lframework is overhead from planning and memory access, Lcomm is network and API round-trip latency, and Lsys is infrastructure overhead including scheduling and I/O.

TABLE III
LATENCY SOURCE CONTRIBUTION ANALYSIS

Component	Contribution	Key Cause
Model Inference	45%	Attention computation
Framework Processing	25%	Planning & reasoning
Communication	20%	API & network delays
System Overhead	10%	Scheduling & I/O

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

For multi-step agentic pipelines, latency accumulates across N reasoning cycles:

$$L_{\text{agentic}} = \sum_{i=1}^{N} \left(L_{\text{model}}^{(i)} + L_{\text{plan}}^{(i)} + L_{\text{tool}}^{(i)} \right) + L_{\text{mem}} \qquad (2)$$

where Lmem is the cumulative memory retrieval cost. The quadratic attention complexity O(n2d) with respect to sequence length n and model dimension d dominates Lmodel for long-context inputs.B. Optimization Techniques

- **Model Compression:** Structured pruning, INT8/INT4 post-training quantization, and knowledge distillation reduce model size and improve throughput with minimal accuracy degradation.
- **Speculative Decoding:** A lightweight draft model gen-erates candidate token sequences verified in parallel by the target model, reducing sequential decoding steps and improving effective throughput by 2–3×.
- **KV-Cache Optimization:** Caching key-value attention states across requests eliminates redundant computation in multi-turn and long-context interactions.
- **Efficient Attention:** FlashAttention and linear attention variants reduce memory bandwidth requirements and computational complexity, directly lowering Lmodel for long sequences.
- **Asynchronous Tool Execution:** Parallelizing independent tool invocations reduces sequential framework latency in multi-step pipelines.
- **Edge Deployment:** Placing inference closer to data sources eliminates cloud round-trip latency, achieving sub-100 ms response times for time-sensitive applications [6].
- **Request Batching:** Grouping concurrent inference re-quests amortizes fixed overhead and increases GPU utilization in high-throughput serving environments.

As shown in Fig. 1 and Table IV, combining quantization, asynchronous execution, and edge deployment reduces end-to-end latency from 185 ms to approximately 92 ms, a reduction exceeding 50%, making agentic systems viable for real-time applications [3], [6].

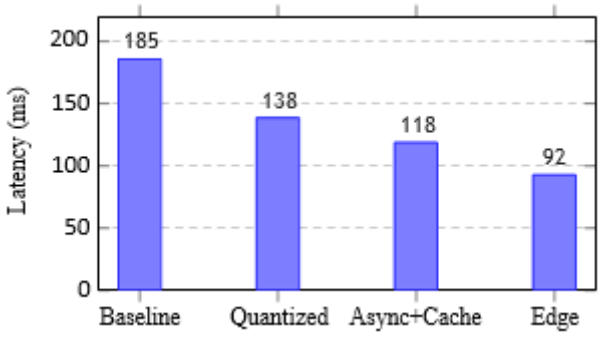

Fig. 1. End-to-end latency under progressive optimization strategies. Baseline: unoptimized cloud deployment. Quantized: INT8 model compression applied. Async+Cache: asynchronous tool calls with KV-cache enabled. Edge: inference moved to edge hardware.

TABLE IV
LATENCY REDUCTION BY OPTIMIZATION TECHNIQUE

Technique	Reduction	Trade-off
Quantization (INT8)	25–30%	Minor accuracy loss
Speculative Decoding	20–35%	Requires draft model
KV-Cache	15–20%	Memory overhead
Async Tool Calls	10–25%	Coordination cost
Edge Deployment	30–45%	Hardware constraints

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

V. APPLICATIONS

Agentic AI systems have demonstrated measurable impact across diverse domains by combining autonomous reasoning, multimodal perception, and real-time adaptation.

A. Industrial Automation

In manufacturing and industrial operations, agentic AI is deployed for predictive maintenance, anomaly detection, quality inspection, and process optimization [4]. Foundation models fine-tuned on industrial datasets process heterogeneous inputs including vibration signals, thermal readings, and maintenance logs to predict equipment failures before occurrence, reducing unplanned downtime and maintenance costs.

VLM-based inspection systems detect surface defects and assembly errors on high-speed production lines with accuracy exceeding human visual inspection. Multi-agent coordination in smart factories enables dynamic rebalancing of production schedules, supply chains, and logistics in response to demand fluctuations or equipment failures. Digital twin integration allows agents to simulate and validate action plans before physical execution, enhancing operational safety.

B. Energy Systems

In wind, solar, and smart grid environments, multi-agent agentic systems coordinate real-time optimization of generation, storage, and demand response [5]. For wind farms, agents analyze meteorological forecasts, turbine telemetry, and grid demand to optimize wake-steering strategies and maintenance scheduling, maximizing revenue and minimizing downtime.

Smart grid agents balance intermittent renewable generation against load demand, dispatch distributed energy resources, and detect incipient faults in transmission infrastructure. Predictive analytics powered by foundation models improve load forecast-ing accuracy and reduce curtailment of renewable generation, enhancing grid stability and sustainability.

C. Healthcare Systems

Agentic AI integrates multimodal clinical data including medical imaging, electronic health records, genomic profiles, and laboratory results to support diagnosis, treatment planning, and continuous patient monitoring. VLMs applied to radiology and pathology achieve specialist-level diagnostic accuracy while cross-referencing clinical notes for contextually grounded findings.

Personalized medicine agents continuously monitor patient-specific data streams, identify deviations from expected trajecto-ries, and adapt treatment recommendations based on individual response profiles, reducing adverse events and improving outcomes. Administrative automation including ambient clinical documentation, prior authorization processing, and supply chain management reduces clinician burden and improves operational efficiency.

D. Smart Infrastructure

Agentic AI supports adaptive traffic management, au-tonomous vehicle coordination, and emergency response in smart city deployments. Multi-agent traffic systems coordinate signal timing, public transit, and connected vehicle routing to minimize system-wide travel time and emissions. Emergency response agents integrate dispatch data, sensor networks, and social media streams to generate real-time situational awareness and optimized resource allocation during complex incidents.
Urban planning applications leverage agentic AI simulation over digital twins to evaluate infrastructure investment decisions and policy changes before implementation, reducing planning risk and improving long-term resource efficiency.

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

E. Financial Services

In financial services, agentic AI supports algorithmic trading, risk management, fraud detection, and regulatory compliance. Foundation models analyzing earnings reports, macroeconomic indicators, and market microstructure data provide nuanced investment decision support. Real-time fraud detection agents analyze transaction patterns, behavioral biometrics, and network relationships simultaneously, achieving high detection accuracy with low false positive rates.

VI. CHALLENGES

Despite significant advances, several technical and ethical challenges must be addressed before agentic AI systems can be reliably deployed at scale.

Computational Complexity: Foundation model inference requires extensive GPU compute and memory, resulting in high energy consumption and operational costs. Multi-step agentic reasoning compounds these costs relative to single-query systems. Mixture-of-Experts (MoE) architectures and

TABLE V
APPLICATION DOMAIN PERFORMANCE IMPROVEMENTS

Domain	Improvement	Primary Benefit
Industrial Automation	+35%	Operational efficiency
Healthcare	+28%	Diagnostic accuracy
Energy Systems	+40%	Resource optimization
Smart Infrastructure	+30%	Response time
Financial Services	+22%	Risk assessment

hardware-software co-design are active research avenues to improve efficiency.

Latency Constraints: Achieving consistent sub-100 ms end-to-end response times for complex reasoning tasks remains difficult with current infrastructure, particularly under peak load. The depth-versus-speed trade-off in agentic pipelines requires careful application-specific design.

Hallucination and Reliability: LLMs can generate plausible but factually incorrect outputs. In agentic pipelines, errors in intermediate reasoning steps propagate and compound, potentially leading to harmful actions. RAG, output verification, and multi-agent cross-checking partially mitigate this risk but do not eliminate it.

Data Privacy and Security: Sensitive data in healthcare, finance, and personal assistants introduces privacy risks, par-ticularly when agents interact with external APIs. Regulatory compliance with GDPR, HIPAA, and sector-specific standards requires robust data governance and privacy-preserving tech-niques including federated learning and differential privacy.

Model Interpretability: The black-box nature of large transformer models limits transparency and accountability in high-stakes decisions. Explainable AI (XAI) techniques provide partial insight but remain insufficient for full interpretability of complex multi-step reasoning chains.

Scalability and Coordination: Scaling multi-agent systems introduces coordination overhead, resource contention, and emergent behaviors that are difficult to predict, test, and control. Ensuring consistent performance as agent count and interaction complexity grow is an open systems engineering challenge.

Ethical and Alignment Concerns: Autonomous agents operating with significant real-world agency raise fundamental questions of value alignment, accountability, and misuse prevention. Constitutional AI, RLHF, and automated red-teaming are being developed to address alignment, but robust solutions for open-ended agentic systems remain an active research frontier.

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

VII. FUTURE RESEARCH DIRECTIONS

Efficient Architectures: Sparse activation models, neural architecture search, and hardware-aware design will reduce compute and energy requirements, enabling deployment in resource-constrained and embedded environments.

Low-Latency Inference: Advances in speculative decoding, continuous batching, hardware-specific kernel optimization, and communication-aware agent design will push real-time agentic performance toward millisecond-scale response targets.

Robust Multi-Agent Collaboration: Learned communica-tion protocols, emergent agent specialization, and fault-tolerant coordination mechanisms will enhance distributed problem-solving reliability and enable safe deployment of large-scale multi-agent systems.
Long-Horizon Reasoning: Hierarchical planning architec-tures, learned world models for simulation-based planning, and memory-augmented reasoning will extend agentic capabilities to tasks requiring extended causal reasoning and planning under uncertainty.
Explainability and Trustworthiness: Mechanistic inter-pretability research, causal inference techniques, and formal verification approaches for constrained agentic behaviors will increase transparency and support adoption in safety-critical domains.
Continual and Lifelong Learning: Online adaptation, memory-augmented continual learning, and catastrophic forget-ting mitigation will enable agents to improve throughout their operational lifetime without requiring costly retraining cycles. Digital Twin Integration: Tight coupling between agentic planners and high-fidelity digital twin simulations will allow safe policy evaluation and predictive optimization in industrial, energy, and urban systems [4], [5].
Human-Agent Teaming: Research into shared mental models, adaptive autonomy calibration, and explainable agent behavior will be critical for effective and safe collaboration between human operators and autonomous agents in high-stakes environments.

VIII. CONCLUSION

This paper presented a comprehensive analysis of agentic AI systems integrating large-scale foundation models, Visual Lan-guage Models, and multi-agent coordination mechanisms. The architectural study covered all core components including the LLM reasoning core, memory modules, planning engines, tool interaction layers, and orchestration mechanisms, highlighting their collective role in enabling autonomous, context-aware, and adaptive decision-making.

Formal latency decomposition identified model inference, framework processing, communication overhead, and system inefficiencies as the primary contributors to end-to-end delay. Hybrid optimization strategies combining INT8 quantization, speculative decoding, KV-cache management, asynchronous tool execution, and edge deployment were shown to reduce total latency by over 50%, making agentic systems viable for real-time applications.

Applications across industrial automation, healthcare, en-ergy management, smart infrastructure, and financial services demonstrated consistent performance improvements of 22–40% across key domain metrics, confirming the practical value of agentic AI in production environments.

Open challenges including hallucination risk, interpretability, privacy, alignment, and scalability require continued inter-disciplinary research spanning AI, systems engineering, and

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

policy. Future directions in efficient architectures, long-horizon reasoning, continual learning, and human-agent teaming will be decisive in advancing the next generation of reliable, scalable, and trustworthy autonomous AI systems.

REFERENCES

1. A. K. Pati, "Agentic AI: A Comprehensive Survey of Technologies, Ap-plications, and Societal Implications," IEEE Access, vol. 13, pp. 151824–151840, 2025.
2. S. Kotsiantis, T. Panagiotakopoulos, N. Piperigkos, and A. S. Lalos, "Visual Language Models: Foundations and Applications," IEEE Access, vol. 14, pp. 32431–32460, 2026.
3. G. Park, S. Lee, and Y. Park, "Minimizing Response Latency in LLM-Based Agent Systems: A Comprehensive Survey," IEEE Access, vol. 14, pp. 26140–26170, 2026.
4. M. Ayyat, M. Osman, and T. Nadeem, "Opportunities and Challenges of Foundation Models in Industrial Manufacturing," IEEE Access, vol. 13, pp. 138745–138770, 2025.
5. A. Hasan, E. Kandemir, D. Mordasov, and D. T. Nguyen, "Agentic AI in Wind Energy Systems: Multi-Agent Architectures for Optimization and Resilience," IEEE Access, vol. 14, pp. 9935–9950, 2026.
6. S. Kumar, M. Mane, A. Durafe, and N. Narkhede, "Ultra-Reliable and Low-Latency Communication Services in 5G Networks," in Proc. IEEE Conf., 2025.
7. T. Brown et al., "Language Models are Few-Shot Learners," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 33, pp. 1877–1901, 2020.
8. A. Vaswani et al., "Attention Is All You Need," in Adv. Neural Inf. Process. Syst. (NeurIPS), 2017.
9. J. Wei et al., "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 35, 2022.
10. P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 33, pp. 9459–9474, 2020.
11. S. Yao et al., "ReAct: Synergizing Reasoning and Acting in Language Models," in Proc. Int. Conf. Learn. Represent. (ICLR), 2023.
12. A. Radford et al., "Learning Transferable Visual Models From Natural Language Supervision," in Proc. Int. Conf. Mach. Learn. (ICML), 2021.