A Hierarchical Multi-Agent Framework for Automated Radar Signal Analysis

Zhao Wang[1], Jiahui Yan[1], Maoguo Gong[1,2,*], Peng Li[1], Hao Li[1]
[1]*Key Laboratory of Collaborative Intelligence Systems, Ministry of Education, School of Electronic Engineering, Xidian University*, Xi'an, China
[2]*College of Artificial Intelligence, Inner Mongolia Normal University*, Hohhot, China
[*]Corresponding author: Maoguo Gong

***Abstract*—Radar signal analysis is pivotal for electromagnetic situational awareness, but faces a critical "perception-reasoning gap." While state-of-the-art deep learning models demonstrate superior performance in deterministic modulation classification, they lack the cognitive capability to perform multi-step reasoning and verify physical consistency. To address this limitation, this paper proposes RadarAgent, a novel hierarchical Multi-Agent framework designed for comprehensive and interpretable signal analysis. Governed by a rigorous Plan-and-Execute cognitive cycle, RadarAgent orchestrates a Supervisor-Worker architecture that integrates a deterministic toolchain with a Retrieval-Augmented Generation (RAG) system for domain knowledge injection. Notably, the toolchain is empowered by our proposed Dual-Scale Radar Transformer (DSRT) to ensure state-of-the-art modulation classification accuracy. Furthermore, a physics-based Critic Agent is introduced to enforce domain constraints, mitigating the hallucination risks inherent in large language models. To evaluate complex reasoning capabilities, we construct RadarQA, a specialized reasoning-intensive benchmark. Experimental results demonstrate that RadarAgent not only significantly outperforms baseline models on the DeepRadar dataset, particularly under low-SNR conditions (-8 dB), but also achieves a robust task success rate of 94.2% on RadarQA. These findings validate the framework's ability to provide fully automated, physically consistent, and interpretable signal analysis reports.**

***Index Terms*—Radar Signal Analysis, Multi-Agent Framework, Large Language Models, Modulation Recognition, Automated Signal Processing**

I. Introduction

Electromagnetic situational awareness constitutes a fundamental capability in modern spectrum monitoring. However, the electromagnetic environment is evolving towards unprecedented complexity, characterized by the proliferation of Low Probability of Intercept (LPI) signals and agile modulation schemes. Traditional analysis paradigms, which rely heavily on manual interpretation by domain experts, are constrained by subjectivity and scalability bottlenecks, rendering them inadequate for real-time processing of massive spectral data.

Data-driven approaches, particularly Deep Learning, have emerged as a powerful alternative. Discriminative models ranging from classical SVMs [1] to advanced architectures such as CNNs [2], [3] and RNNs [4] have achieved remarkable success in modulation classification. Despite their high accuracy, these methods fundamentally operate as "black boxes." They excel at mapping signal features to predefined labels but lack the cognitive capability to deduce the physical implications of signal parameters or to perform multi-step reasoning. For instance, a standard classifier might identify a waveform as Linear Frequency Modulation (LFM) but cannot verify whether the extracted bandwidth and pulse width satisfy physical consistency constraints, nor can it adaptively invoke analysis tools for novel signal types.

The emergent paradigm of Large Language Model (LLM)-based Agents offers a transformative solution. By replacing rigid prediction heads with a flexible "Plan-and-Execute" cognitive cycle, agents can decompose complex problems into executable sub-tasks. While this hierarchical agentic framework has demonstrated potential in domains such as maritime traffic control [5] and healthcare simulation [6], its application to radar signal analysis remains unexplored. A significant gap persists in integrating the probabilistic perception of neural networks with the deterministic rigor of signal processing algorithms.

To bridge this gap, we propose RadarAgent, a novel hierarchical multi-agent framework designed for physically consistent signal analysis. Unlike monolithic models, RadarAgent adopts a transparent Supervisor-Worker architecture. It orchestrates a specialized perception engine alongside a deterministic toolchain to achieve both high-precision recognition and logical reasoning.

Our main contributions are summarized as follows:

- We propose a Multi-Agent Framework that integrates a generative planner, a deterministic executor, and a physics-based critic. This architecture enables an automated, interpretable workflow that surpasses traditional single-modal deep learning models.
- We introduce the Dual-Scale Radar Transformer (DSRT), a specialized perception backbone featuring a dual-attention mechanism. It serves as the core perception engine, achieving state-of-the-art performance in modulation recognition tasks.
- We construct RadarQA, a comprehensive benchmark designed to evaluate the multi-step reasoning capabilities of

This project is supported by National Natural Science Foundation of China No. 62306221, 62576259.

signal analysis agents, addressing the lack of reasoning-oriented test suites in the radar domain.
- Empirical evaluations demonstrate that RadarAgent not only outperforms state-of-the-art baselines on the public DeepRadar dataset but also achieves a robust task success rate of 94.2% on the RadarQA benchmark, validating its effectiveness in complex analytical scenarios.

II. Related Works

A. Deep Learning for Automatic Modulation Recognition

Automatic Modulation Recognition (AMR) is a pivotal component of non-cooperative signal processing. Early approaches relied on expert-crafted features, which proved brittle in low-SNR environments. The advent of deep learning revolutionized this field. [7] proposed a baseline LSTM architecture that effectively captures temporal dependencies in I/Q sequences. More recently, Transformer-based models [8] have achieved state-of-the-art (SOTA) performance by leveraging self-attention mechanisms to model global signal correlations. However, these discriminant models operate as "black boxes"—they provide classification labels but cannot generate interpretable analysis reports or reason about the physical implications of the signal parameters, a gap our work aims to bridge.

B. Large Language Models for Time-Series Analysis

The formidable reasoning capabilities of Large Language Models (LLMs) have prompted their adaptation to continuous data domains. [9] introduced Time-LLM, a framework that reprograms LLMs for time-series forecasting by aligning numerical embeddings with textual tokens. In the wireless domain, [10] explored utilizing LLMs for network optimization and protocol generation. Despite these advancements, applying LLMs directly to raw radar signals remains problematic. The distinct modality gap between electromagnetic signals and natural language often leads to severe hallucinations when models attempt to perform precise mathematical operations without external aids [11]. To overcome the limitations of standalone LLMs, the research community has shifted towards agentic workflows. The ReAct paradigm [12] synergizes reasoning and acting, enabling models to interleave thought generation with external tool retrieval. To handle long-horizon tasks, Plan-and-Solve strategies [13] and Hierarchical architectures like HuggingGPT [14] utilize a controller to decompose complex objectives into executable sub-goals. Furthermore, to mitigate error propagation, reflexive mechanisms such as Reflexion [15] introduce verbal reinforcement learning, allowing agents to self-correct based on feedback. Our RadarAgent builds upon these foundational architectures but introduces a specialized "Physics Critic" to enforce domain-specific constraints, ensuring that the reasoning process adheres to electromagnetic laws.

C. Vertical Domain Agents

Recently, specialized agents tailored for vertical domains have emerged. In remote sensing, RS-Agent [16] integrates visual foundation models with interpretative tools to automate satellite image analysis. Similarly, VTS-LLM [5] enhances situational awareness in vessel traffic services through knowledge-retrieval mechanisms. In the broader context of complex system control, frameworks like LLM+P [17] and Chain of Code [18] have demonstrated the efficacy of decoupling high-level planning from low-level deterministic execution. However, a significant gap remains in the radar domain: no existing framework seamlessly integrates raw I/Q signal processing with physics-constrained reasoning. RadarAgent fills this void by establishing a loop specifically designed for electromagnetic signal analysis.

III. METHODOLOGY

A. RadarAgent Framework

To rigorously formulate the sequential decision-making process in radar signal analysis, we cast the RadarAgent framework as a Partially Observable Markov Decision Process (POMDP) [19]. The problem is defined by the tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, \mathcal{O} \rangle$, where the Supervisor Agent acts as the global controller, responsible for high-level task planning and inter-agent coordination. It orchestrates the Planner to decompose complex queries and the Executor to interact with the environment. Specifically:

- $\mathcal{S}$ (State Space): Represents the latent signal characteristics (e.g., true modulation parameters, SNR) and the agent's internal cognitive state.
- $\mathcal{A}$ (Action Space): Comprises the set of discrete tool invocations ($a \in \mathcal{A}_{tool}$) and natural language reasoning steps defined in the Toolchain.
- $\mathcal{T}$ (Transition): Denotes the deterministic state evolution triggered by tool execution.

1) Hierarchical Decomposition Strategy: Given a complex user query q, the Supervisor's objective is to maximize the probability of generating the correct analysis report y. Unlike direct generation $P(y|q)$, RadarAgent adopts a latent variable model, where the reasoning process is mediated by a dynamic plan $\mathcal{P}$:

$$P(y|q) = \sum_{\mathcal{P}} \underbrace{P(y|\mathcal{P}, \mathcal{K}, q)}_{\text{Execution}} \cdot \underbrace{P(\mathcal{P}|q)}_{\text{Planning}} \tag{1}$$

where $\mathcal{P} = \{g_1, g_2, \ldots, g_K\}$ represents the sequence of sub-goals generated by the Planner Agent, and $\mathcal{K}$ denotes the domain knowledge and tool library. This decomposition reduces the complexity of the reasoning space from exponential to linear with respect to the task horizon [13].

2) The Plan-and-Execute Cognitive Cycle: The Planner and Executor agents operate in a coupled cognitive loop. At time step t, the state s_t encompasses the global context and the history of execution results H_{t-1}.

1) **Planning Policy (π_{plan}):** The Planner generates the next instruction i_t based on the current feedback:

$$i_t \sim \pi_{plan}(i_t|q, H_{t-1}) \tag{2}$$

2) **Execution Policy (π_{exec}):** The Executor maps the natural language instruction i_t to a specific tool invocation $a_t \in \mathcal{A}_{tool}$:

$$a_t, o_t = \text{ToolChain}(i_t) \tag{3}$$

where o_t is the observation derived from the tool output. This iterative process aligns with the ReAct paradigm [12], synergizing reasoning traces with action trajectories to prevent error propagation.

3) Physics-Informed Critical Evaluation: To enforce physical consistency, we introduce a *Physics-Constraint Filter* acting on the observation space. Unlike standard POMDPs, where agents accept all observations unconditionally, RadarAgent incorporates a verification function $V(\cdot)$ derived from electromagnetic domain knowledge Φ. Addressing the practical measurement tolerances, we replace the rigid binary indicator with a soft constraint energy function. The valid observation probability is formally gated as:

$$P(o_t|s_t, a_t) \propto \exp\left(-\lambda \cdot \max(0, \Delta_{err} - \tau)\right) \tag{4}$$

where Δ_{err} is the relative measurement error, τ is the acceptable tolerance threshold, and λ is a penalty decay factor. This formulation mathematically ensures that the final analysis trajectory satisfies the kinematic constraints of the radar domain, effectively pruning hallucinatory reasoning branches while maintaining robustness against minor measurement noise.

Implementation Details & Parameters: For reproducibility, the execution policy π_{exec} is driven by few-shot prompt templates (detailed in the Appendix). In the RAG-enhanced module, the hybrid search strategy combines dense vector similarity and sparse lexical matching (BM25) with weighting parameters of $\alpha = 0.7$ and $1 - \alpha = 0.3$, respectively, ensuring both semantic relevance and precise terminology recall.

Closed-Loop Error Correction. To fully bridge the perception-reasoning gap, RadarAgent implements a reflexive closed-loop mechanism. When an observation fails the physics validation, the system does not simply halt. Instead, the Critic Agent generates a natural language error diagnostic. This diagnostic is appended to the Shared Analysis State and fed back to the Planner. The Planner is then forced to initiate a Correction Phase, dynamically adjusting tool parameters or invoking alternative reasoning paths up to a maximum of N_{retry} times. This feedback loop ensures that perception errors are iteratively resolved through physics-informed reasoning rather than silently propagated.

Toolchain. To anchor the agent's cognitive reasoning in rigorous empirical evidence, we establish a specialized signal analysis toolchain, the specifications of which are detailed in Table I. This framework operates in two distinct phases to ensure both accuracy and physical plausibility:

1) *Primary Analysis Phase (Executor):* This phase employs a hybrid pipeline combining classical signal processing with deep learning to extract actionable features. The `extract_parameters` and `analyze_pri` tools utilize statistical estimation and envelope autocorrelation to derive fundamental metrics, including SNR, bandwidth, and Pulse Repetition Interval (PRI) [20], [21]. Simultaneously, the classification tool deploys the proposed DSRT to provide fine-grained probabilistic recognition of 23 complex modulation types.
2) *Physical Validation Phase (Critic):* To mitigate the risk of hallucinatory reasoning common in LLMs, we incorporate a physics-based verification layer. The Critic Agent utilizes rule-based tools to enforce domain constraints. By validating the kinematic consistency of Doppler shifts ($f_d \approx 2v/\lambda$) and flagging potential bandwidth-PRF ambiguities, this mechanism ensures that the agent's analysis remains grounded in the physical laws governing radar propagation.

TABLE I
SPECIFICATION OF THE SIGNAL ANALYSIS TOOLCHAIN

Tool Primitive	Functional Specification & Logic
	Phase I: Perception & Feature Engineering
`Neural Recognizer`	**Logic:** Deploys the DSRT backbone to estimate the probability distribution over 23 modulation types from raw I/Q sequences. **Output:** Top-k predicted classes with confidence scores ({`type: "LFM", conf: 0.98`}).
`Waveform Analyzer`	**Logic:** A hybrid pipeline combining Hann-windowed FFT and envelope autocorrelation. Extracts statistical parameters and derived features, including **Time-Bandwidth Product (TBP)** and Duty Cycle. **Thresholds:** Dynamic noise floor estimation ($\mu + 2\sigma$).
	Phase II: Physics & Knowledge Reasoning
`Kinematic Validator`	**Logic:** Validates signal parameters against the platform's kinematic profile. Checks for **Doppler Consistency** ($\lvert f_{meas} - f_{exp} \rvert < \delta$) and flags potential Range-Doppler ambiguities. **Role:** Acts as a physics-based guardrail to reject hallucinated classifications.
`Threat Retriever`	**Logic:** Retrieves tactical threat profiles from the knowledge base using a **hybrid search** (Dense Vector + BM25). Matches extracted fingerprints to radar operational modes. **Output:** Retrieved context for intent deduction.

RAG-Enhanced Knowledge System. Integrating domain expertise, we have built Retrieval-Augmented Generation (RAG). This system is based on the 47 volumes of the "National Defense Electronics Information Technology Series". The agents dynamically implement the mixed search strategy [22], which integrates both the traditional BM25 [23] and a vector-based search [24] approach to rank the document set. The reranked model [25] is adopted to select the optimal context as the input. After obtaining satisfactory reference model analysis results, the agent translates the complete model results based on the template into a fact-based professional report.

B. The Dual-Scale Radar Transformer (DSRT)

To address the limitations of traditional deep learning models in capturing both local transient features and global long-range dependencies in radar signals, we propose the *Dual-Scale Radar Transformer (DSRT)*. As illustrated in Fig. 2, this hierarchical architecture integrates a hybrid convolutional

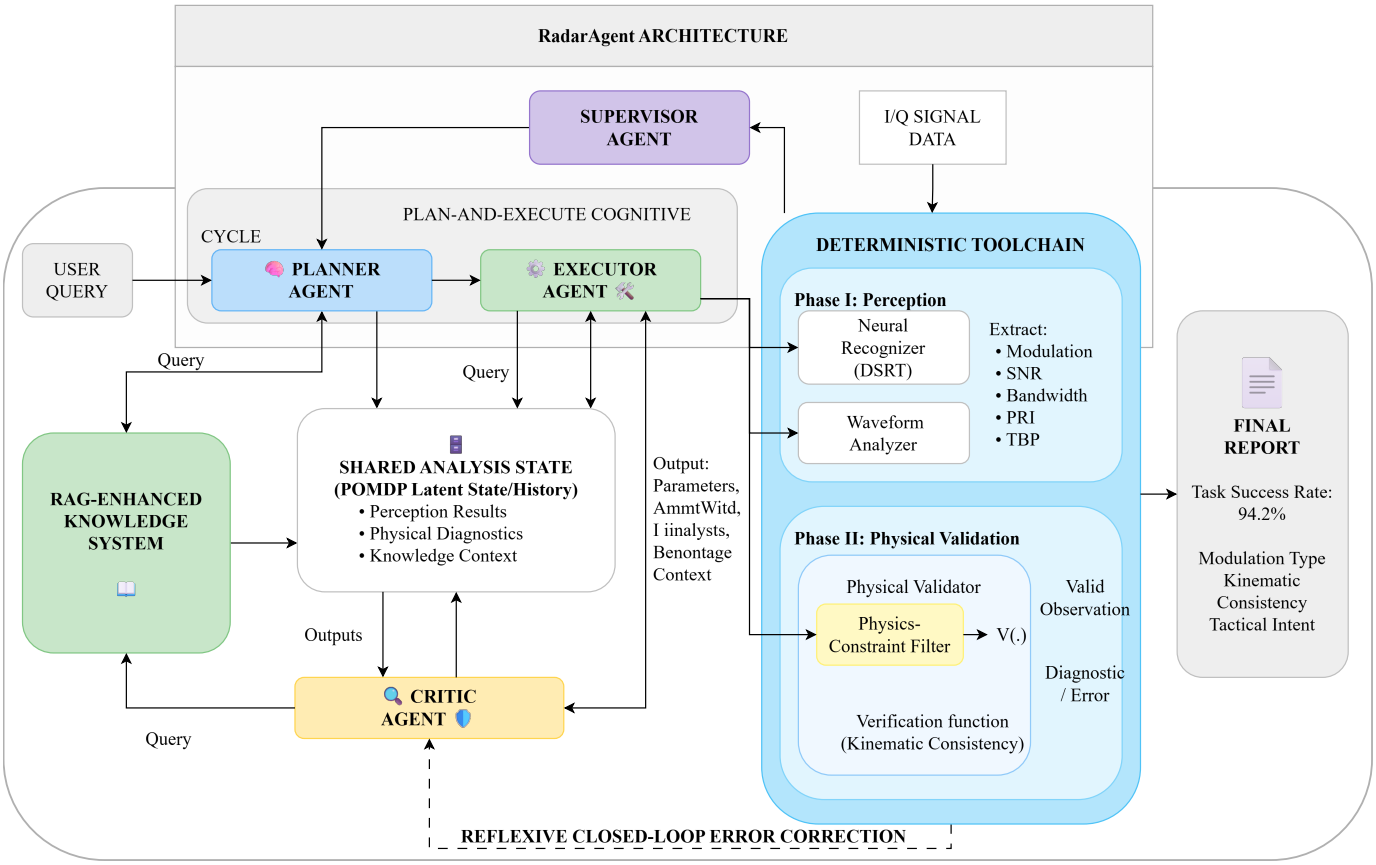

Fig. 1. The architecture of RadarAgent, a hierarchical multi-agent framework. A central supervisor orchestrates a planner, executor, and critic in a cognitive loop to process a user query and raw signal data. The agents leverage two key external components: a deterministic toolchain for empirical analysis and an RAG knowledge system for domain expertise. All findings are centralized in a Shared Analysis State. ensuring a robust and verifiable analysis that culminates in a final report.

stem with a multi-stage encoder that employs a specialized dual-attention mechanism to resolve the unique time-frequency characteristics of electromagnetic signals.

1) Hybrid Convolutional Stem: Given a raw I/Q signal input $\mathbf{X} \in \mathbb{R}^{B\times 2\times L}$, where B is the batch size and L is the sequence length, the stem module performs initial feature extraction and downsampling. It consists of two cascaded 1D convolutional layers. The first layer uses a kernel size of 7 and a stride of 2 to capture broad spectral features, while the second layer uses a kernel size of 3 and a stride of 2 to refine these features. This reduces the sequence length to $L/4$ and projects the channel dimension to an embedding size C. To preserve sequence order information essential for time-series analysis, we add absolute positional encodings $\mathbf{P}$ to the feature map $\mathbf{Z}_0$:

$$\mathbf{Z}_0 = \text{ConvStem}(\mathbf{X}) + \mathbf{P} \tag{5}$$

2) Hierarchical Encoder Stages: The core of the network consists of three hierarchical stages. We adopt a pyramid structure where the feature resolution is progressively reduced while the channel dimension is increased. Between stages, a *Patch Merging* layer reduces the sequence length by a factor of 2 and doubles the channel dimension using a linear projection layer, defined as:

$$\mathbf{Z}_{out} = \text{LayerNorm}(\mathbf{W}_{pm} \cdot \text{Reshape}(\mathbf{Z}_{in})) \tag{6}$$

where $\mathbf{W}_{pm} \in \mathbb{R}^{2C\times 2C}$ is the trainable projection matrix.

3) Dual-Scale Attention Mechanism: Radar signals exhibit a unique dual-scale temporal structure: rapid phase variations within a pulse (micro-scale) and repetitive patterns across pulse trains (macro-scale). To capture these conflicting features efficiently, we alternate between two distinct attention mechanisms within each encoder stage: Contextual Transformer (CoT) Attention [26] for local transients and Cross-Covariance Attention (XCA) [27] for global dependencies.

4) Micro-Scale: Contextual Transformer Attention: Standard self-attention often neglects the rich contextual information among neighboring keys, which is critical for identifying intra-pulse modulation features such as chirp rates or phase shifts. We employ CoTAttention to mine static and dynamic contexts. For an input feature map $\mathbf{Z}$, we first define keys $\mathbf{K}$ and values $\mathbf{V}$. The static context $\mathbf{K}^{(s)}$ is learned via a grouped convolution with kernel size k to enforce local spectral continuity:

$$\mathbf{K}^{(s)} = \text{Conv1d}_{k\times k}(\mathbf{Z}) \tag{7}$$

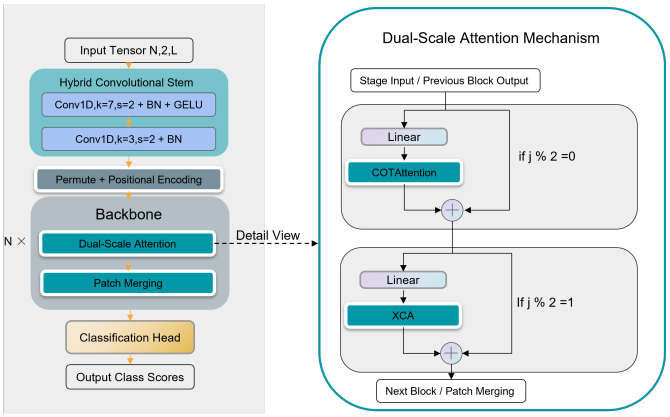

Fig. 2. Architecture of the Dual-Scale Radar Transformer (DSRT). It integrates a Hybrid Convolutional Stem, three Hierarchical Encoder Stages with alternating CoT and XCA attention mechanisms, and a Classification Head for modulation recognition.

The dynamic context is generated by concatenating the static context with the input query and passing it through two consecutive convolutional layers W_θ and W_δ:

$$\mathbf{A} = \text{Softmax}\big(W_\delta(\sigma(W_\theta([\mathbf{K}^{(s)}, \mathbf{Z}])))\big) \tag{8}$$

where σ denotes the ReLU activation function. The final output is the fusion of the static context and the dynamic attention-weighted values:

$$\mathbf{Z}_{CoT} = \mathbf{K}^{(s)} + \mathbf{A} \otimes \mathbf{V} \tag{9}$$

This mechanism effectively acts as a learnable time-frequency filter, isolating transient pulses common in radar modulation.

5) Macro-Scale: Cross-Covariance Attention (XCA): To model global dependencies with linear complexity, we utilize Cross-Covariance Attention (XCA). Unlike standard self-attention, which operates along the sequence length dimension (computing an $L \times L$ map), XCA operates along the feature dimension. Given queries $\mathbf{Q}$, keys $\mathbf{K}$, and values $\mathbf{V}$, we compute the attention map by interacting transposed keys and queries:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathbf{V} \cdot \text{Softmax}\left(\frac{\mathbf{K}^T\mathbf{Q}}{\tau}\right) \tag{10}$$

where τ is a learnable scaling parameter. This global interaction ensures that the model can correlate features across the entire pulse duration, capturing the high-level semantic identity of the emitter without the quadratic computational cost.

C. The RadarQA Reasoning Benchmark

We developed **RadarQA**, a reasoning-intensive benchmark designed to evaluate the cognitive depth of radar agents. Constructing such a benchmark presents a unique challenge: real-world data is often unlabeled, while synthetic data lacks physical context. We address this through a three-stage construction pipeline grounded in the public DeepRadar dataset [7], ensuring that all reasoning tasks are built upon verified, high-fidelity I/Q signal sequences.

Addressing the absence of environmental context in raw I/Q samples, our core innovation is Scenario Metadata Injection. We explicitly augment the dataset by binding a synthetic physical profile to each signal sample. By simulating the reception of propagating radar signals and recording the relevant parameters, we empower the agent's Critic module to execute validation checks, moving beyond simple pattern matching.

To scale this process and ensure linguistic diversity, we adopt an LLM-driven synthetic data generation approach. Rather than relying on rigid templates, we define structured "seed scenarios" that combine signal features with the injected metadata. These seeds are then processed by a foundation model with specific instructions to evolve them into natural language queries of varying complexity. This method produces questions that mimic the nuanced and often ambiguous nature of real-world commander intents, ranging from simple parameter inquiries to complex tactical threat assessments.

The resulting RadarQA dataset comprises 200 query-signal pairs stratified into five difficulty levels. Level 1 and Level 2 focus on fundamental perception, requiring the agent to retrieve signal attributes or perform single-step classification. Level 3 and Level 4 introduce multi-hop reasoning, where the agent must correlate signal features with external domain knowledge, such as deducing radar function from the Time-Bandwidth Product. Level 5 represents the highest complexity, demanding long-horizon planning and report generation. Crucially, these complex tasks explicitly require the agent to cross-verify signal parameters against the injected scenario metadata to detect anomalies. Each sample is paired with a "Golden Reasoning Trace," providing a ground truth sequence of tool calls and logical deductions for fine-grained process evaluation.

IV. Experimental Setup

A. Datasets & Metrics

We evaluate our framework on two datasets. For fundamental signal perception, we use the public DeepRadar benchmark [7], comprising 23 modulation types across SNRs from -12 dB to $+20$ dB. Performance is measured via Accuracy and Micro F1-score.

To evaluate multi-step analytical capabilities, we constructed RadarQA, a reasoning-intensive dataset with 200 query-signal pairs explicitly augmented with scenario metadata. Stratified into five difficulty levels, RadarQA tests the agent's cognition using four metrics: Task Success Rate (SR) , Reasoning Accuracy , Tool Misuse Rate (TMR), and RAGAS-based [28] Faithfulness and Context Precision.

V. Results and Discussion

A. Performance on Modulation Recognition

We conducted a comprehensive evaluation of the signal recognition on the DeepRadar dataset. Table II benchmarks our proposed framework against a range of models, including the classic DeepRadar LSTM [7], ResNet-1D [29], and MCformer [30], as well as the state-of-the-art hybrid architecture CDAT-32 [31]. Our DSRT model establishes a new performance

Case A: Level-3 Reasoning Task (Multi-hop Inference)

Signal ID: `DR_Sample_2309_LFM.iq` (SNR: +12dB)

User Query: "Analyze the modulation parameters of this signal. Based on its time-bandwidth product, what is the likely operational mode?"

Agent Trace: nosep, leftmargin=1.5em

1) `classify_modulation(target)` → `"LFM"` (Conf: 0.98)
2) `extract_parameters(target)` → {`BW: 150 MHz, PW: 10 μs, PRI: 200 μs`}
3) *Self-Correction:* "Bandwidth is high. Calculating Time-Bandwidth Product (TBP)."
4) `calc_TBP(150MHz, 10us)` → `1500` (High TBP)
5) `rag_query("High TBP LFM applications")` → "High resolution SAR imaging."

Final Report: "The signal is identified as **Linear Frequency Modulation (LFM)**. With a bandwidth of 150 MHz and a Time-Bandwidth Product of 1500, it indicates a high-resolution pulse compression waveform, typically associated with **Synthetic Aperture Radar (SAR)** mapping modes."

Case B: Level-5 Comprehensive Analysis (Physics-Constrained Report Generation)

Signal ID: `Intercept_XJ_09.iq` (SNR: -4dB)
Scenario Meta: {*Platform: "Airborne Unknown", Velocity: ≈ 680 m/s (Mach 2), Range: 45 km, Aspect: "Head-on"*}

User Query: "Perform a threat assessment. Verify if the signal's kinematic features are consistent with the reported approach velocity. Generate a tactical report."

Agent Trace: *(Step 1: Planner decomposes task into 4 sub-goals...)*

leftmargin=1.5em, noitemsep, topsep=0pt

2) **Plan:** 1. Signal Characterization → 2. Kinematic Validation → 3. Threat Library Matching → 4. Report.
3) `classify_modulation(target)` → `"Costas_Code"`
4) `extract_parameters(target)` → {`fc: 9.6 GHz, BW: 50MHz`}
5) *Planner Thought:* "Need precise Doppler for Goal 2. Standard extraction insufficient."
6) `fine_grain_doppler(target)` → `+44.5 kHz`
[...] *(Steps 7-12: SNR verification and PRI analysis omitted)*
13) *Critic Check:* "Validating Doppler against Mach 2 profile."
14) **`check_doppler(meas_fd=44.5k, ref_v=680)`** → {`valid: True, error: 2.1%`}
15) `rag_retrieve("X-band Costas + Mach 2")` → *[3 candidates found]*
16) *Reasoning:* "Candidate A (Weather Radar) rejected due to velocity. Candidate B (FCR) matches."

Final Report: "**[THREAT CONFIRMED]** ... (Report generated at Step 22)"

Fig. 3. Comparison of reasoning complexity in RadarQA. **Top (Level 3):** Multi-hop reasoning combining signal parameters (TBP) with domain knowledge. **Bottom (Level 5):** A comprehensive analysis task requiring scenario metadata integration. Note the specific invocation of the `check_doppler` tool by the Critic Agent to validate physical consistency before generating the threat report.

standard, particularly under challenging low-SNR conditions. At -8 dB, DSRT achieves an accuracy of 88.2%, significantly surpassing the SOTA CDAT-32 (84.5%) by 3.7%. As SNR improves to 0 dB and 4 dB, all models exhibit increased accuracy. While CDAT-32 achieves parity at 4 dB (97.5%), DSRT demonstrates superior robustness in the lower SNR regime.

Although the performance of all models degrades in the extreme noise regime, RadarAgent effectively mitigates this drop by correcting ambiguous samples through multi-tool collaborative analysis. Consequently, it achieves the highest Overall Average Accuracy of 94.8% across the full testing range (-12 dB to +20 dB), validating its robustness in complex electromagnetic environments."

B. Impact of Foundation Models

To quantify module contributions, we conducted a systematic ablation study (Table III). The *No-Tools* variant exhibited the most significant degradation, particularly in L5 tasks (33% drop), confirming that deterministic tools are indispensable for grounding abstract reasoning in precise signal parameters to prevent hallucinations. While the *No-RAG* configuration maintained robust baseline performance (87.3%) due to the LLM's internal knowledge, the full agent's ∼15% lead in L5 tasks highlights the critical role of external retrieval for accessing proprietary domain details. Furthermore, the *No-Plan* variant's performance collapses to 62.1% in complex scenarios, revealing the limitations of Chain-of-Thought (CoT), validating the Planner's necessity for maintaining logical consistency in long-horizon interactions (>20 steps).

C. Evaluation of Agent Reasoning Capabilities

Finally, we investigated the agent's performance against standard prompting baselines and across different core Large Language Models (LLMs). To ensure statistical rigor and address generation randomness, all reasoning experiments on RadarQA were independently repeated over 5 trials, with results reported as mean ± standard deviation.

As shown in Table IV, standard prompting frameworks (using GLM-4.5) like Zero-Shot CoT [32] and ReAct [12] struggle significantly in complex electromagnetic analysis, achieving success rates of only 44.0% and 71.3%, respectively.

TABLE II
PERFORMANCE COMPARISON WITH BASELINES ON DEEPRADAR DATASET

Method	Backbone	Accuracy at Specific SNRs				Overall Metrics†	
		-8 dB	-4 dB	0 dB	4 dB	Avg. Acc.	Micro F1
DeepRadar LSTM	RNN	62.1%	78.5%	91.2%	96.3%	82.3%	0.82
ResNet-1D	CNN	68.4%	75.6%	91.5%	96.1%	86.1%	0.86
MCformer	Attn	75.4%	86.3%	92.8%	96.9%	88.5%	0.88
CDAT-32 (SOTA)	Hybrid	84.5%	92.1%	94.4%	**97.5%**	90.5%	0.90
DSRT (Ours)	Attn+Conv	**88.2%**	**93.1%**	**95.6%**	97.4%	**92.4%**	**0.92**
RadarAgent (Ours)	DSRT	**90.3%**	**94.6%**	**96.9%**	**98.2%**	**94.8%**	**0.95**

†Overall metrics are computed over the full SNR range (-12 dB to +20 dB).

TABLE III
ABLATION STUDY OF COMPONENT CONTRIBUTION (SUCCESS RATE)

Method	Difficulty Level			Overall
	L1-L2	L3-L4	L5	Avg
No-Tools	75.5±1.8%	68.3±2.1%	42.5±3.5%	62.1±2.4%
No-RAG	96.4±0.9%	90.7±1.2%	74.8±2.1%	87.3±1.4%
No-Plan	97.5±0.8%	88.2±1.5%	62.1±2.8%	82.6±1.7%
RadarAgent	**98.8±0.5%**	**94.5±0.8%**	**89.2±1.4%**	**94.2±0.9%**

TABLE IV
IMPACT OF BASELINES AND CORE LLMS ON REASONING

Method / Core LLM	Success Rate	Reas. Acc.	Tool Misuse
Prompting Baselines (using GLM-4.5)			
Zero-Shot CoT	44.0±3.1%	40.5±3.5%	–
ReAct	71.3±2.5%	68.2±2.8%	15.4±1.8%
RadarAgent with Different LLMs			
DeepSeek-r1	86.5±1.8%	85.2±1.9%	7.8±1.2%
Qwen3-Max	90.1±1.2%	89.5±1.5%	4.5±0.8%
GLM-4.5	**94.2±0.9%**	**93.8±0.8%**	**2.1±0.4%**

TABLE V
QUALITY EVALUATION OF THE RAG KNOWLEDGE SYSTEM USING RAGAS METRICS

Metric (Avg)	Context Precision	Context Recall	Answer Relevancy	Faithfulness
L3 (Reasoning)	0.94	0.98	0.97	0.99
L4 (Multi-hop)	0.91	0.95	0.94	0.97
L5 (Report)	0.88	0.92	0.93	0.96
Overall Avg	**0.91**	**0.95**	**0.95**	**0.97**

The high tool misuse rate (15.4%) in ReAct highlights the necessity of our hierarchical physics-constrained architecture.

Furthermore, inference capability is highly correlated with the model's architectural focus—particularly the trade-off between "pure inference" and "precise tool execution." We compared three representative architectures equipped with our framework: the inference model DeepSeek-r1 [33], the balanced general-purpose model Qwen3-Max [34], and GLM-4.5 [35]. DeepSeek-r1 demonstrated strong baseline performance (86.5% success rate) but exhibited a higher tool misuse rate (7.8%), indicating that while its internal logic is robust, it occasionally struggles to strictly adhere to complex JSON tool schemas. Qwen3-Max struck a better balance (90.1% success rate). Ultimately, GLM-4.5, optimized for agentic workflows, achieved the SOTA success rate of 94.2±0.9% with the lowest tool misuse rate (2.1±0.4%), confirming its superior stability in translating reasoning traces into deterministic actions.

D. Quality of Retrieval-Augmented Generation

To rigorously evaluate the reliability of the external knowledge injection, we employed the RAGAS framework to assess the RAG module across four dimensions: Context Precision, Context Recall, Answer Relevancy, and Faithfulness. The results are detailed in Table V. The system achieves an outstanding Faithfulness score of 0.97, confirming that the agent's reasoning is strictly grounded in the retrieved domain literature, effectively minimizing hallucinations. The Context Recall (0.95) indicates that the retriever successfully locates relevant threat profiles and physical formulas in 95% of cases. We observe a slight decline in Context Precision (0.94 $\rightarrow$ 0.88) as tasks progress from L3 to L5. This is expected, as Level 5 tasks involve complex, multi-faceted queries that may retrieve broader, partially noisy contexts. However, the Answer Relevancy remains robust (0.93) even at Level 5, demonstrating that the LLM's internal attention mechanism effectively filters noise to generate responses that directly address the user's analytical needs.

VI. CONCLUSION AND FUTURE WORK

This paper proposes RadarAgent, a hierarchical multi-agent framework that bridges the gap between neural perception and logical reasoning in radar signal analysis. By integrating the DSRT with a physics-constrained cognitive architecture,

we achieve a robust reasoning success rate of 94.2% on the proposed RadarQA benchmark.

Crucially, RadarAgent aligns with the core principles of Human-Centered AI (HCAI). By transparently exposing the epistemic trajectory (from signal parameter extraction to physics-based validation) and generating interpretable reports, the framework significantly reduces the cognitive load on radar operators and fosters human-machine trust in mission-critical environments.

While this pioneering framework demonstrates high efficacy, we acknowledge certain limitations that pave the way for future research. First, transitioning from the synthetic-augmented RadarQA to real-world, complex electromagnetic environments remains a critical challenge. Second, to meet the stringent latency requirements of real-time spectrum monitoring, future iterations will focus on optimizing the framework for edge deployment through model quantization and the integration of low-resource LLMs. Ultimately, we envision incorporating a human-in-the-loop feedback mechanism to continuously refine the agent's tool selection and physics constraints based on operator expertise.

REFERENCES

[1] T. Wan, K. Jiang, Y. Tang, Y. Xiong, and B. Tang, "Automatic lpi radar signal sensing method using visibility graphs," *IEEE Access*, vol. 8, pp. 159 650–159 660, 2020.

[2] T. Erpek, T. J. O'Shea, Y. E. Sagduyu, Y. Shi, and T. C. Clancy, "Deep learning for wireless communications," in *Development and Analysis of Deep Learning Architectures*. Springer, 2019, pp. 223–266.

[3] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, 2018.

[4] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, 2018.

[5] S. Sun, L. Zhao, M. Deng, and X. Fu, "Vts-llm: Domain-adaptive llm agent for enhancing awareness in vessel traffic services through natural language," 2025. [Online]. Available: https://arxiv.org/abs/2505.00989

[6] J. Li, Y. Lai, W. Li, J. Ren, M. Zhang, X. Kang, S. Wang, P. Li, Y.-Q. Zhang, W. Ma, and Y. Liu, "Agent hospital: A simulacrum of hospital with evolvable medical agents," 2025. [Online]. Available: https://arxiv.org/abs/2405.02957

[7] V. Clerico, J. González-López, G. Agam, and J. Grajal, "Lstm framework for classification of radar and communications signals," in *2023 IEEE Radar Conference (RadarConf23)*, 2023, pp. 1–6.

[8] Z. Ma, S. Fang, Y. Fan, G. Li, and H. Hu, "An efficient and lightweight model for automatic modulation classification: A hybrid feature extraction network combined with attention mechanism," *Electronics*, vol. 12, no. 17, 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/17/3661

[9] M. Jin, S. Wang, L. Ma, Z. Chu, J. Y. Zhang, X. Shi, P.-Y. Chen, Y. Liang, Y.-F. Li, S. Pan, and Q. Wen, "Time-llm: Time series forecasting by reprogramming large language models," 2024. [Online]. Available: https://arxiv.org/abs/2310.01728

[10] L. Liang, H. Ye, Y. Sheng, O. Wang, J. Wang, S. Jin, and G. Y. Li, "Large language models for wireless communications: From adaptation to autonomy," 2025. [Online]. Available: https://arxiv.org/abs/2507.21524

[11] Y. Zhang, Y. Li, L. Cui, D. Cai, L. Liu, T. Fu, X. Huang, E. Zhao, Y. Zhang, Y. Chen *et al.*, "Siren's song in the ai ocean: A survey on hallucination in large language models," *arXiv preprint arXiv:2309.01219*, 2023. [Online]. Available: https://arxiv.org/abs/2309.01219

[12] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, "React: Synergizing reasoning and acting in language models," *arXiv preprint arXiv:2210.03629*, 2022.

[13] L. Wang, W. Xu, Y. Lan, Z. Hu, Y. Lan, R. K.-W. Lee, and E.-P. Lim, "Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models," 2023. [Online]. Available: https://arxiv.org/abs/2305.04091

[14] Y. Shen, K. Song, X. Tan, D. Li, W. Lu, and Y. Zhuang, "Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face," 2023. [Online]. Available: https://arxiv.org/abs/2303.17580

[15] N. Shinn, F. Cassano, E. Berman, A. Gopinath, K. Narasimhan, and S. Yao, "Reflexion: Language agents with verbal reinforcement learning," 2023. [Online]. Available: https://arxiv.org/abs/2303.11366

[16] W. Xu, Z. Yu, B. Mu, Z. Wei, Y. Zhang, G. Li, and M. Peng, "Rs-agent: Automating remote sensing tasks through intelligent agent," 2025. [Online]. Available: https://arxiv.org/abs/2406.07089

[17] B. Liu, Y. Jiang, X. Zhang, Q. Liu, S. Zhang, J. Biswas, and P. Stone, "Llm+p: Empowering large language models with optimal planning proficiency," *arXiv preprint arXiv:2304.11477*, 2023.

[18] C. Li, J. Liang, A. Zeng, X. Chen, K. Hausman, D. Sadigh, S. Levine, L. Fei-Fei, F. Xia, and B. Ichter, "Chain of code: Reasoning with a language model-augmented code emulator," 2024. [Online]. Available: https://arxiv.org/abs/2312.04474

[19] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[20] M. I. Skolnik, *Radar Handbook*, 3rd ed. New York, NY, USA: McGraw-Hill Education, 2008.

[21] M. A. Richards, *Fundamentals of Radar Signal Processing*, 3rd ed. New York, NY, USA: McGraw-Hill Education, 2022.

[22] C. S. Mala, G. Gezici, and F. Giannotti, "Hybrid retrieval for hallucination mitigation in large language models: A comparative analysis," 2025. [Online]. Available: https://arxiv.org/abs/2504.05324

[23] X. Chen and S. Wiseman, "Bm25 query augmentation learned end-to-end," 2023. [Online]. Available: https://arxiv.org/abs/2305.14087

[24] Y. A. Malkov and D. A. Yashunin, "Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 4, pp. 824–836, 2020.

[25] Y. Zhang, M. Li, D. Long, X. Zhang, H. Lin, B. Yang, P. Xie, A. Yang, D. Liu, J. Lin, F. Huang, and J. Zhou, "Qwen3 embedding: Advancing text embedding and reranking through foundation models," *arXiv preprint arXiv:2506.05176*, 2025.

[26] Y. Li, T. Yao, Y. Pan, and T. Mei, "Contextual transformer networks for visual recognition," 2021. [Online]. Available: https://arxiv.org/abs/2107.12292

[27] A. El-Nouby, H. Touvron, M. Caron, P. Bojanowski, M. Douze, A. Joulin, I. Laptev, N. Neverova, G. Synnaeve, J. Verbeek, and H. Jegou, "Xcit: Cross-covariance image transformers," 2021. [Online]. Available: https://arxiv.org/abs/2106.09681

[28] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "Ragas: Automated evaluation of retrieval augmented generation," 2025. [Online]. Available: https://arxiv.org/abs/2309.15217

[29] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, p. 168–179, Feb. 2018. [Online]. Available: http://dx.doi.org/10.1109/JSTSP.2018.2797022

[30] W. Han, T. Zhu, L. Chen, H. Ning, Y. Luo, and Y. Wan, "Mcformer: Multivariate time series forecasting with mixed-channels transformer," *IEEE Internet of Things Journal*, vol. 11, no. 17, pp. 28 320–28 329, 2024.

[31] Z. Yi, H. Meng, L. Gao, Z. He, and M. Yang, "Efficient convolutional dual-attention transformer for automatic modulation recognition," *Applied Intelligence*, vol. 55, no. 3, pp. 1–16, 2024. [Online]. Available: https://doi.org/10.1007/s10489-024-06202-6

[32] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 22 199–22 213.

[33] DeepSeek-AI, "Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning," 2025. [Online]. Available: https://arxiv.org/abs/2501.12948

[34] Q. Team, "Qwen3 technical report," 2025. [Online]. Available: https://arxiv.org/abs/2505.09388

[35] GLM Team, A. Zeng, X. Lv, Q. Zheng, Z. Hou, B. Chen *et al.*, "GLM-4.5: Agentic, reasoning, and coding (ARC) foundation models," 2025. [Online]. Available: https://arxiv.org/abs/2508.06471