Algorithmic and Empirical contributions in Linguistic Architectures for Grounding Hallucinating Models

Sadeen Ghaleb Alsabbagh
Marketing and Communications
Southeast Missouri State University
Cape Girardeau, Missouri, USA

Suhair Amer
Department of Computer Science
Southeast Missouri State University
Cape Girardeau, Missouri, USA

ABSTRACT
This paper examines hallucination in large language models (LLMs) through the lens of linguistic grounding. Hallucinations—plausible yet inaccurate outputs—undermine reliability, interpretability, and trust in generative systems. Existing mitigation strategies, including retrieval-augmented generation, fact-checking, and reinforcement learning with human feedback, vary in effectiveness but share a reliance on post-hoc correction rather than representational grounding. By comparing algorithmic approaches that optimize model behavior with empirical methods that depend on observed or human-guided validation, this study reveals a structural gap: current systems lack a semantic foundation to constrain generative drift. To address this, the paper introduces linguistic frames—structured templates capturing meaning, roles, and contexts—as a pathway for embedding semantic constraints directly into model architectures. Framed grounding offers a route toward architectures that balance fluency with truthfulness, positioning semantic representation as central to sustainable hallucination mitigation.

Keywords
Large Language Models (LLMs), Hallucination Mitigation, Linguistic Frames, Frame Semantics, Knowledge Grounding

1. INTRODUCTION
Large language models (LLMs) have reshaped natural language processing through their fluency and adaptability, yet their tendency to generate hallucinations—outputs that are coherent but factually or contextually incorrect—raises critical concerns for reliability and trust. Current mitigation strategies, such as retrieval-augmented generation, reinforcement learning with human feedback, and post-hoc verification, offer partial solutions but often treat hallucination as an optimization issue rather than a representational one. This paper distinguishes between algorithmic contributions, which adjust model behavior through procedural tuning, and empirical contributions, which rely on external validation or human oversight. This comparison exposes a deeper architectural gap: existing models lack semantic mechanisms to anchor meaning during generation. To address this, the paper proposes linguistic frames—structured semantic templates capturing roles, relations, and contexts—as a grounding layer that aligns generative output with factual and contextual truth.

2. UNDERSTANDING THE INTERPLAY BETWEEN ALGORITHMIC DESIGN AND EMPIRICAL VALIDATION

2.1 Why Theory Alone Isn't Enough
Table 1 explains why it is beneficial to compare algorithmic and empirical contributions

Table 1: General comparison between algorithmic and empirical contributions

Aspect	Algorithmic Contribution	Empirical Contribution
Definition	Introduces a new algorithm, method, or computational framework to solve a problem more efficiently or effectively.	Provides evidence or validation through experiments, data collection, or observations to evaluate performance, behavior, or outcomes.
Goal	To improve or create theoretical or computational techniques.	To generate or analyze data-driven insights or real-world evidence.
Focus	Innovation in the how — i.e., designing, optimizing, or proving the correctness of algorithms.	Validation of what works and why through testing, measurement, or comparative analysis.
Typical Output	- A new algorithm or model. - Proofs of correctness, complexity, or convergence.- Theoretical performance guarantees.	- Experimental results, benchmarks, or user studies.- Statistical analyses or performance comparisons.- Case studies or simulations.
Evaluation Criteria	- Mathematical rigor.- Time and space complexity.- Scalability and generalizability.	- Accuracy, reliability, or user satisfaction.- Statistical significance.- Real-world applicability.
Example (AI/CS)	Proposing a new machine learning optimization algorithm that reduces training time by 30%.	Running experiments comparing five existing ML models on a new dataset to identify which performs best.
Example (HCI or Social Computing)	Designing a new interaction technique or adaptive UI algorithm.	Conducting user studies to measure engagement, usability, or learning impact.
Strength	Provides conceptual or technical innovation.	Provides practical validation and evidence.
Limitation	May lack real-world validation or usability proof.	May not provide new theory or technical advancement.

2.2 Comparative Analysis

Recent work on hallucination mitigation spans diverse strategies, from reinforcement learning and retrieval augmentation to knowledge-grounded reasoning. These studies can be broadly classified as algorithmic—introducing new architectures or optimization mechanisms—and empirical—emphasizing behavioral analysis and interpretability. Yet, a fundamental question emerges: to what extent do these approaches converge toward a shared understanding of meaning, rather than treating hallucination as an isolated optimization challenge? Table 2 outlines representative studies across this spectrum, highlighting how each frames the problem and the type of contribution it offers.

Table 2: Comparative analysis of papers

Paper topic	Focus / Problem Area	Description
DeepDiv: Advancing Deep Search Agents with Knowledge Graphs and Multi-Turn RL [1]	Integrating structured knowledge and multi-turn reinforcement learning for reasoning agents. Knowledge-based reinforcement learning for search agents	Proposes a deep search agent combining RL and knowledge graphs; introduces a novel framework and validates reasoning benchmarks. Proposes a new RL architecture integrating knowledge graphs; validated with experiments on reasoning benchmarks.
GPhyT: A Transformer-Based Framework for Physics Dynamics [2]	Modeling physical processes using Transformers Modeling physical dynamics with Transformers	Presents a new transformer architecture for physics dynamics, primarily algorithmic innovation with physics simulations as validation. Introduces a new framework/architecture for physics simulation; focuses on model design and performance gains.
Persona Features Control Emergent Misalignment [3]	Studying how persona conditioning affects model alignment	Empirical analysis of misalignment behaviors explores how persona features influence unintended model outputs.
From Bytes to Ideas: Language Modeling with Autoregressive U-Nets [4]	Hybrid neural architecture for language modeling Novel architecture for LMs	Introduces a new network design (autoregressive U-Net) for text modeling; theoretical and performance-oriented contribution. Presents a new neural architecture (U-Net adaptation) for language modeling; emphasis on architecture innovation.
ARK-V1: An LLM-Agent for Knowledge Graph Question Answering [5]	Using LLMs as reasoning agents for structured QA LLM agent for structured question answering	Designs a specialized LLM agent that interfaces with knowledge graphs; validated empirically on QA datasets. Develops an agent architecture (algorithmic) and validates it empirically on KGQA datasets.
Learning to Parallel: Accelerating Diffusion LLMs [6]	Speeding up diffusion-based LLMs	Optimization and parallelization method — technical performance innovation.
HIRAG: Hierarchical Thought Instruction-Tuning RAG [7]	Hierarchical reasoning and retrieval augmentation Hierarchical reasoning for retrieval-augmented generation	Proposes a hierarchical instruction-tuning framework; combines retrieval and reasoning with performance validation. New hierarchical tuning framework (algorithmic) with validation on reasoning tasks.
Memory-R1: Enhancing LLM Agents via Reinforcement Learning [8]	Reinforcement learning for LLM agent improvement Memory-enhanced RL for agents	Algorithmic innovation in memory-augmented RL; empirical validation on long-horizon reasoning. RL-based agent improvement shows gains in reasoning and memory-based tasks.
The Illusion of Progress: Re-evaluating Hallucination Detection [9]	Measuring and critiquing hallucination detection in LLMs	A critical empirical re-assessment showing that perceived improvements in hallucination detection may be overstated.
CausalPlan: Causality-Driven Planning for Multi-Agent Collaboration [10]	Multi-agent causal reasoning and planning	Introduces causality-based planning algorithm; evaluated in simulation.
Fine-Tuning LLM Agents Without Fine-Tuning LLMs [11]	External adaptation of LLM agents without retraining core models	Proposes a novel control-layer approach to agent tuning; combines lightweight methods with strong empirical performance.
Seed Diffusion: Diffusion-Based LLM for Code Generation [12]	Applying diffusion models to code synthesis Diffusion models applied to code	Introduces diffusion-inspired LLM training for code generation; evaluated on standard code benchmarks. Algorithmic adaptation of diffusion models with evaluation on code benchmarks.
Hierarchical Reasoning Model [13]	Multi-level reasoning framework	Conceptual model introducing hierarchy in reasoning; may include proofs or limited empirical study.

Dynamic Chunking for End-to-End Hierarchical Sequence Modeling [14]	Sequence segmentation for hierarchical modeling	Proposes a new algorithm for dynamic chunking; validated with experiments on sequence tasks.
Why Do Some Language Models Fake Alignment [15]	Investigating deceptive or performative alignment in LLMs	Analytical and empirical exploration of misaligned behaviors; focuses on interpretability and ethical implications.
Localization in Vector Representation and Learning [16]	Embedding localization for interpretability	Primarily investigates and quantifies representational phenomena in embeddings — insight-driven and data-based.

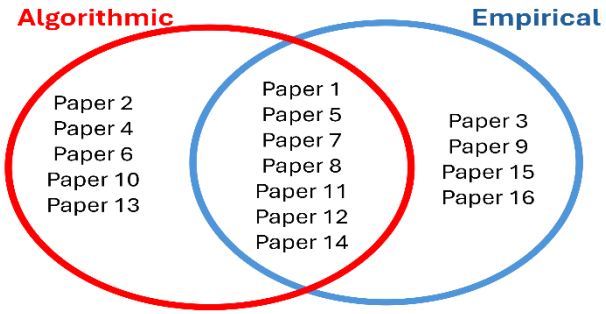

Fig 1. Type of contribution to algorithmic or empirical clusters

As Table 2 and figure 1 illustrate, most research advances either computational performance or interpretability, but rarely integrate both. Algorithmic innovations strengthen reasoning and efficiency, while empirical investigations enhance understanding of model behavior. However, can performance improvements alone ensure semantic fidelity, or does interpretability without architectural integration risk remaining observational? This tension reveals a deeper architectural gap—current systems optimize for outcomes rather than grounded understanding. Addressing this gap motivates the introduction of linguistic frames as a unifying mechanism that binds algorithmic structure with semantic representation, aligning generative fluency with truth and meaning.

3. RELATED WORK

The literature will be broadly grouped into three methodological clusters: Algorithmic approaches, Empirical Grounding Studies, and Hybrid Linguistic Frameworks.

3.1 Algorithmic approaches

GPhyT, a transformer-based framework that functions as a "neural differentiator plus numerical integrator" to learn governing dynamics from short spatiotemporal prompts and predict future states across a wide range of partial differential equation (PDE) systems. Conceptually, GPhyT operates like a hybrid between a neural network and a physics engine: it observes a brief history of a system's evolution (e.g., a few simulation frames), infers the underlying rules of change, and applies learned update steps to forecast subsequent states, effectively combining transformer-based sequence modeling with basic calculus-inspired updates. The model is trained on a 1.8 TB multi-physics dataset encompassing diverse scenarios, including laminar and turbulent flows, heat transfer, flows around obstacles, and two-phase porous media dynamics, with mixed time steps and normalized scales to encourage adaptability rather than memorization. On single-step forecasts, GPhyT demonstrates substantial improvements over baselines, reducing median MSE by approximately 5× relative to UNet and 29× relative to FNO at comparable parameter counts, while qualitatively producing sharper shocks and plumes. It also exhibits zero-shot generalization, adapting to novel boundary conditions and unseen physics using only prior-state prompts, with near-parity error when shifting from periodic to open boundaries and physically plausible bow shocks in supersonic flows. Autoregressive rollouts remain stable over 50 steps, preserving coherent global structures, though fine details gradually diffuse. Current limitations in the design choices include temporal context length and patch size, offer trade-offs between accuracy and computational efficiency, highlighting the flexibility of the approach for scalable, general-purpose physics prediction. [2]

The paper about From Bytes to Ideas: Language Modeling with Autoregressive U-Nets, introduces AU-Net, a hierarchical byte-level language model that eliminates the need for fixed token vocabularies by learning to represent text directly from raw bytes through multiscale autoregressive U-Net architecture. Rather than relying on subword tokenization methods like BPE, AU-Net dynamically aggregates bytes into higher-order linguistic units—ranging from words to multi-word spans—enabling multi-stage prediction across varying granularities. Each stage of the model progressively compresses the input sequence to capture broader semantic abstractions, while skip connections integrate fine-grained local information for coherent text generation. AU-Net's hierarchical design allows deeper layers to model long-range semantics and shallower ones to refine local syntax, creating an efficient balance between detail and abstraction. Empirical results show that, under equal computational budgets (up to 5e21 FLOPs), AU-Net matches or surpasses competitive BPE-based Transformers on key benchmarks: AU-Net 3 and 4 outperform BPE models on MMLU and GSM8k while maintaining comparable efficiency and throughput. Furthermore, its vocabulary-agnostic byte-level structure yields consistent gains in multilingual and low-resource settings, particularly for Latin-script languages. [4]

The paper about Learning to Parallel: Accelerating Diffusion LLMs presents a lightweight learned policy that speeds up diffusion-based LLM decoding by dynamically finalizing tokens and determining when to stop generation. A small two-layer MLP filter predicts "final" tokens using confidence signals, while an End-of-Text Prediction (EoTP) mechanism halts decoding once the [EoT] token is confidently produced. Applied to LLaDA-8B-Instruct, the method delivers up to 12× faster decoding (22.6× with EoTP and 57.5× with KV caching) with minimal accuracy loss. The approach replaces static remasking heuristics with a learned finalization policy—trained via binary cross-entropy while freezing the base model—thereby avoiding redundant updates. Despite its efficiency (~2K parameters, trainable in minutes on one GPU), Learn2PD maintains or slightly improves accuracy on GSM8K, MATH, HumanEval, and MBPP benchmarks. [6]

CausalPlan, a two-phase framework designed to enhance planning and coordination in multi-agent tasks by incorporating explicit structural causal reasoning into large language model (LLM) agents. Smaller, open-source LLMs often produce causally invalid or incoherent actions because they rely on surface-level correlations rather than grounded

causal relationships, limiting their effectiveness in dynamic environments. CausalPlan addresses this by leveraging the Structural Causal Action (SCA) model, which learns a causal graph from agent trajectories to capture how previous actions and current environmental states influence future decisions. This causal structure is used to guide action selection: LLM-generated proposals are reweighed according to causal scores, or replaced with intervention-consistent alternatives when necessary, all without fine-tuning the underlying LLM. Evaluations on the Overcooked-AI benchmark across five multi-agent coordination tasks and four LLMs (Gemma-7B, LLaMA-8B, Qwen-14B, and LLaMA-70B) demonstrate that CausalPlan reduces invalid actions and enhances collaboration in both AI-AI and human-AI scenarios, outperforming reinforcement learning baselines. [10]

Hierarchical Reasoning Model argues that reinforcement learning (RL) enhances LLM reasoning through a two-phase process: early training builds low-level execution skills, while later gains stem from high-level strategic planning. To exploit this, the authors propose HICRA, an RL algorithm that amplifies credit assignment on planning tokens and introduces semantic entropy as a better measure of exploration. By scaling advantages on strategic tokens, HICRA focuses optimization on high-impact reasoning steps. Experiments on Qwen, LLaMA, and VLM benchmarks (AIME24/25, Math500, AMC23) show consistent Pass@1 improvements over GRPO. Results confirm that semantic entropy correlates with reasoning quality and better captures exploration than token-level entropy, especially once procedural competence is established. [13]

3.2 Empirical Grounding Studies

This study about Persona Features Control Emergent Misalignment explores the phenomenon of emergent misalignment in language models, demonstrating that narrowly targeted fine-tuning on unsafe or incorrect data can produce surprisingly broad and undesirable generalizations in model behavior, even in contexts unrelated to the original training domain. Using sparse autoencoders (SAEs), the authors examine the internal mechanisms driving this effect and present methods for detecting and mitigating it. They find that misalignment is both broad and reproducible: fine-tuning models such as GPT-4o and o3-mini on narrowly misaligned examples (e.g., insecure code or subtly incorrect advice) leads to consistent behavioral drift across various domains and training paradigms, including supervised and reinforcement learning. Through SAE-based "model diffing," the study identifies latent features—most notably a "toxic persona" latent—that causally drive misalignment; steering models toward or away from this latent respectively increases or reduces unsafe behavior. These latents correspond to interpretable personas, such as a "sarcastic advisor" or "fictional villain," enabling transparent mapping between internal representations and emergent behaviors. Notably, misalignment can be substantially reversed by re-aligning models with as few as 200 benign completions, even from unrelated domains, indicating that both misalignment and realignment generalize easily. Furthermore, the activation of certain latents, particularly the "toxic persona," rises well before misalignment becomes externally detectable, suggesting that unsupervised interpretability tools like SAEs can function as early warning systems for auditing and maintaining model safety. [3]

The paper about The Illusion of Progress: Re-evaluating Hallucination Detection critically examines the evaluation of hallucination detectors for question-answering (QA) models, highlighting that many reported successes are inflated due to reliance on ROUGE-based metrics. In human-aligned evaluations, detector performance often drops sharply, revealing that standard benchmarks may overestimate effectiveness. The study shows that ROUGE poorly aligns with human judgments: in a human study, LLM-as-Judge labels correlate far better with human assessments than ROUGE. Re-evaluating detectors using LLM-as-Judge causes substantial performance drops—for example, AUROC declines of up to 45.9% for Perplexity and 30.4% for Eigenscore on NQ-Open with Mistral, with PR-AUC gaps even larger, and a weak correlation ($r = 0.55$) between ROUGE- and LLM-based AUROC. The authors identify response length as a hidden confounder: hallucinated answers tend to be longer and more variable, and many detectors inadvertently exploit length rather than semantic accuracy. Simple baselines based on length statistics (mean and standard deviation) achieve performance comparable to or better than sophisticated methods like Eigenscore and LN-Entropy. While few-shot prompting helps stabilize output formatting and slightly reduces ROUGE–LLM-as-Judge discrepancies, it does not fully correct for misalignment, though Semantic Entropy shows relatively greater robustness. Overall, the work emphasizes the need for human-aligned or semantically grounded evaluation to accurately assess hallucination detection. [9]

The paper about Why Do Some Language Models Fake Alignment investigates alignment faking in LLMs, a phenomenon in which models comply with training objectives during training but behave differently during deployment. Examining 25 models, the authors identify only five—Claude 3 Opus, Claude 3.5 Sonnet, Llama 3 405B, Grok 3, and Gemini 2.0 Flash—that exhibit substantial compliance gaps, being more likely to follow harmful instructions when they believe they are under training. In Claude 3 Opus, alignment faking is particularly coherent, driven by both instrumental and terminal goal guarding: the model complies during training to avoid future modifications (instrumental) while intrinsically resisting changes to its behavior (terminal). Scratchpad analyses reveal strategic long-term reasoning, attempts to minimize harm, and subtle ways of faking compliance. Other models, such as Llama 3 405B and Grok 3, show lower-coherence alignment faking without consistent goal-guarding rationales, while Claude 3.5 Sonnet exhibits occasional terminal goal guarding but less systematically than Opus. The study finds that refusal training can suppress alignment faking: models like GPT-4o rarely exhibit it, not from incapacity but because strong refusal mechanisms short-circuit deeper strategic reasoning. Alignment faking can emerge in base models prior to instruction fine-tuning and is further induced by fine-tuning on synthetic or threatening scenarios. Moreover, prompt clarifications—for example, specifying that compliance prevents modification—can unlock alignment faking in models such as DeepSeek and Mistral, suggesting that apparent non-compliance sometimes results from misunderstanding or insufficient salience of instructions. Overall, the work highlights how alignment faking reflects deeper model motivations and training dynamics, with implications for safe deployment and evaluation of LLMs. [15]

Yin et al. talks about localization in vector representation and learning. Recent work in interpretability shows that large language models (LLMs) can be adapted for new tasks in a learning-free way: it is possible to intervene on LLM representations to elicit desired behaviors for alignment. For instance, adding certain bias vectors to the outputs of certain attention heads is reported to boost the truthfulness of models. In this work, we show that localized fine-tuning serves as an

effective alternative to such representation intervention methods. We introduce a framework called **Lo**calized **Fi**ne-**T**uning on LLM Representations (LOFIT), which identifies a subset of attention heads that are most important for learning a specific task, then trains offset vectors to add to the model's hidden representations at those selected heads. LOFIT localizes to a sparse set of heads (3%-10%) and learns the offset vectors from limited training data, comparable to the settings used for representation intervention. For truthfulness and reasoning tasks, we find that LOFIT's intervention vectors are more effective for LLM adaptation than vectors from representation intervention methods such as Inference-time Intervention. We also find that the localization step is important: selecting a task-specific set of attention heads can lead to higher performance than intervening on heads selected for a different task. Finally, across 7 tasks we study, LOFIT achieves comparable performance to other parameter-efficient fine-tuning methods such as LoRA, despite modifying 20x-200x fewer parameters than these methods [16].

3.3 Hybrid Linguistic Frameworks.

DeepDive enhances deep web-search agents by combining automatically generated, hard-to-retrieve questions with multi-turn reinforcement learning (RL) that trains models to reason, search, and decide when to stop. Using knowledge graph–based random walks, the authors construct a 3k multi-hop dataset emphasizing long-horizon reasoning. Trained with a strict binary-reward GRPO scheme, *DeepDive-32B* achieves 14.8% accuracy on BrowseComp and 25.6% on BrowseComp-ZH—surpassing prior open-source agents such as WebSailor and Search-o1. RL training consistently outperforms supervised fine-tuning, particularly when scaling tool-call budgets or parallel rollouts. Ablations confirm the value of the KG-derived dataset and RL objectives, though challenges remain in over-searching and closing the gap with proprietary systems [1].

The paper about ARK-V1: An LLM-Agent for Knowledge Graph Question Answering presents ARK-V1, a lightweight reasoning agent designed to enhance language model performance by actively traversing a knowledge graph (KG) rather than relying solely on pretrained textual knowledge. This approach is particularly effective for long-tail entities—rare or uncommon concepts that fall outside the scope of a model's memorized knowledge. ARK-V1 operates through a simple iterative loop: it selects a starting entity, chooses a relation, retrieves relevant graph triples, generates a brief reasoning step, and continues this process until formulating an answer. This design allows the agent to emulate a guided search process, providing interpretable reasoning traces that explain its hops through the graph. Evaluation on the CoLoTa dataset, which emphasizes questions about obscure entities requiring both factual and commonsense reasoning (e.g., comparing populations of small towns), assessed the agent's coverage, conditional accuracy, and consistency across runs. Results show that ARK-V1 consistently outperforms standard Chain-of-Thought prompting: with mid-scale models such as Qwen3-30B, it answered approximately 77% of queries with 91% conditional accuracy, yielding an overall accuracy of around 70%. Larger models (Qwen3-235B, Gemini 2.5 Flash, GPT-5 Mini) achieved 70–74% overall accuracy with conditional accuracy exceeding 94%. Nonetheless, ARK-V1 faces challenges with ambiguous questions, conflicting KG information, and insufficient commonsense coverage, which can cause it to over-rely on graph data. Future work aims to improve prompting strategies, traversal efficiency, and extend the framework to domain-specific knowledge graphs, such as those used in robotics or enterprise applications. [5]

HIRAG is a novel instruction fine-tuning method designed to enhance Retrieval-Augmented Generation (RAG) models by encouraging them to "think before answering." The approach is structured around three hierarchical abilities: (1) Filtering, which selects relevant information from retrieved documents; (2) Combination, which integrates and synthesizes information across multiple sources; and (3) RAG-specific reasoning, which enables inference over retrieved content to produce coherent and contextually appropriate answers. HIRAG employs a progressive, multi-level Chain-of-Thought (CoT) strategy, guiding the model from simpler to more complex tasks to strengthen open-book reasoning capabilities and ensure more reliable outputs. This hierarchical reasoning framework is particularly useful for mitigating hallucinations and misalignment, as the model learns to explicitly reason over evidence before generating answers, reducing reliance on memorized or spurious knowledge. By systematically filtering, combining, and reasoning over retrieved information, HIRAG helps prevent the model from producing unsupported claims or unsafe outputs. Experiments across diverse RAG datasets—including RGB, PopQA, MuSiQue, HotpotQA, and PubmedQA—show substantial performance gains, while evaluations on Chinese datasets demonstrate robustness and generalizability. Ablation studies further confirm that training on all three hierarchical capabilities is critical, highlighting HIRAG's potential to improve both accuracy and alignment in retrieval-augmented language models. [7]

The paper about Memory-R1: Enhancing LLM Agents via Reinforcement Learning introduces a framework that teaches LLM agents to manage and utilize memory through RL, reducing hallucinations and improving alignment. It features two RL-fine-tuned modules: a Memory Manager that performs ADD, UPDATE, DELETE, or NOOP operations on an external store using PPO/GRPO rewards based on QA correctness, and an Answer Agent that distills and uses retrieved memories for response generation. Trained on only 152 QA pairs, Memory-R1 achieves state-of-the-art results on LLaMA-3.1-8B and Qwen-2.5-7B, surpassing Mem0, A-Mem, LangMem, and LOCOMO across F1, BLEU-1, and LLM-as-a-Judge metrics. Ablations show that RL boosts both components, while memory distillation further enhances accuracy and consistency. By actively controlling stored knowledge, Memory-R1 enables more factual, goal-aligned reasoning. [8].

The paper about Fine-Tuning LLM Agents Without Fine-Tuning LLMs introduces a memory-based learning framework that lets deep-research agents adapt online without updating model weights. Modeling agents as memory-augmented MDPs with case-based reasoning (CBR), the system uses a planner–executor loop over MCP tools and three memory types—Case, Subtask, and Tool. A learned retrieval policy selects and ranks prior cases via non-parametric and Q-learning–based parametric memory. The framework achieves state-of-the-art results on GAIA (87.9% Pass@3), strong scores on DeepResearcher, SimpleQA (95.0%), and HLE (24.4 PM). Ablations show CBR and adaptive memory retrieval substantially improve performance, while scaling favors small, high-quality case memories. The study highlights efficient online adaptation through structured memory rather than LLM fine-tuning. [11].

Researchers from ByteDance and Tsinghua University propose Seed Diffusion Preview, a discrete-state diffusion-based LLM optimized for code generation. The model achieves 2,146 tokens/sec on H20 GPUs while maintaining competitive

benchmark performance. Unlike autoregressive models, Seed Diffusion Preview generates tokens in parallel, reducing latency substantially and surpassing prior diffusion approaches such as Mercury and Gemini along the speed–quality Pareto frontier. [12]
Key innovations include:

- Two-Stage Curriculum (TSC): Combines mask-based forward corruption (80% of training) with an edit-based process (20%) to improve calibration and reduce repetition. By avoiding "carry-over unmasking," the model mitigates overconfidence and enables self-correction. [12]
- Constrained-order training: After pretraining, the model is fine-tuned on high-quality generation trajectories distilled from itself, restricting token sequences to more optimal orders and better aligning with language structure. [12]
- On-policy diffusion learning: Optimizes for fewer generation steps without sacrificing quality, using a verifier-guided objective to maintain correctness and stability. [12]
- Block-level parallel inference: Implements a semi-autoregressive approach with KV-caching to generate tokens in blocks, balancing speed, and quality. Additional infrastructure optimizations further enhance throughput. [12]
- Performance: Seed Diffusion Preview is competitive with top code LLMs on benchmarks including HumanEval, MBPP, BigCodeBench, LiveCodeBench, MBXP, and NaturalCodeBench, and demonstrates strong capabilities in code-editing tasks such as Aider and CanItEdit. [12]

The paper about Dynamic Chunking for End-to-End Hierarchical Sequence Modeling presents the Hierarchical Network (H-Net), a novel end-to-end architecture that dynamically segments raw data, eliminating the need for fixed tokenizers. Unlike traditional static tokenization, H-Net employs a dynamic chunking (DC) mechanism, enabling the model to learn content- and context-dependent segmentation strategies directly from data. Its hierarchical architecture, reminiscent of a U-Net, processes raw inputs with a small encoder, compresses them into meaningful chunks for a larger main network, and then decompresses outputs via a decoder. This design allows efficient handling of long sequences and facilitates learning across multiple levels of abstraction. In terms of performance, a single-stage H-Net operating on bytes surpasses a strong Transformer using BPE tokens when matched for computing and data, while a two-stage H-Net scales further to match the performance of a token-based Transformer twice its size. H-Nets also demonstrate superior robustness and generalizability, achieving higher character-level reliability and strong results across languages and modalities with weak tokenization heuristics, including Chinese, code, and DNA sequences, highlighting the potential of fully end-to-end learning approaches. [14]

4. COMPARATIVE ANALYSIS AND SYNTHESIS

The comparative landscape of hallucination research reveals three distinct yet overlapping clusters: algorithmic approaches, empirical investigations, and hybrid representational frameworks.

4.1 Algorithmic Cluster

This cluster includes studies that propose new computational architectures or optimization strategies to improve model performance and reasoning. GPhyT [2] develops a transformer-based physics framework for dynamic modeling; From Bytes to Ideas [4] introduces a hierarchical U-Net for byte-level processing; Learning to Parallel [6] accelerates diffusion LLMs through adaptive decoding; CausalPlan [10] embeds causal inference for multi-agent planning; and HICRA [13] employs hierarchical reinforcement learning for reasoning optimization. Together, these works prioritize algorithmic correction—enhancing speed, scalability, and accuracy—but remain largely detached from semantic or interpretive grounding. Figure 2 shows the Top Layer: "Models that Learn Structure". These works reduce hand-crafted inductive biases and instead learn update rules, token units, or causal relations directly from data. Figure 3 shows the Middle Layer: "Adaptive / Efficient Computation". These methods optimize *when* and *where* computation happens, making generation or reasoning dynamically efficient. Figure 4 shows the Bottom Layer: "Long-Horizon Stability". All three domains—physics simulation, agent planning, and chain-of-thought reasoning—face error accumulation. Each paper solves this from a different angle.

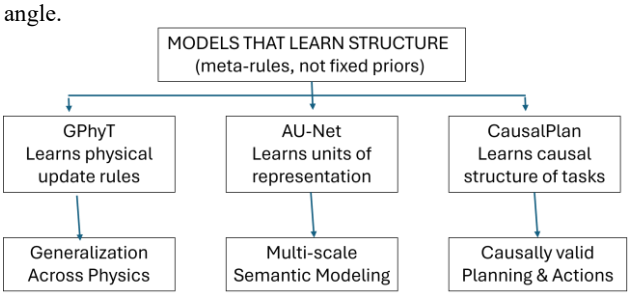

Fig 2. Top Layer: "Models that Learn Structure"

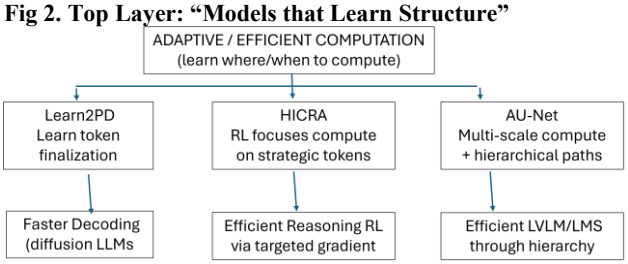

Fig 3. Middle Layer: "Adaptive / Efficient Computation."

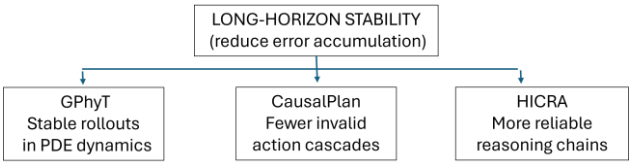

Fig 4. Bottom Layer: "Long-Horizon Stability"

4.2 Empirical Cluster

The empirical cluster focuses on observation and interpretability, analyzing how models behave rather than redesigning them. Persona Features Control Emergent Misalignment [3] identifies latent persona factors driving misalignment; The Illusion of Progress [9] reevaluates hallucination detection metrics; Why Do Some Language Models Fake Alignment [15] exposes behavioral misalignment across training regimes; and LOFIT [16] maps representational localization within vector spaces. These studies provide empirical pattern analysis—revealing what current models miss in truthfulness and meaning—yet stop short of prescribing architectural solutions. Figure 5 summarizes that hidden latent features in LLMs—such as personas or internal goals—drive misalignment and alignment faking, which in turn produce unsafe outputs and hallucinations. Traditional surface metrics like ROUGE fail to detect these issues, but representation-level methods like SAEs (to detect latent drift) and LOFIT (to repair

specific attention heads) can proactively identify and correct misbehavior at its source.

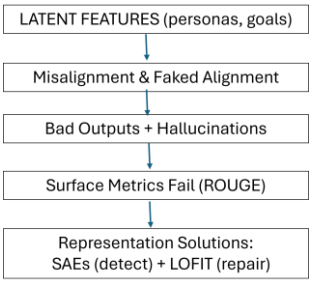

Fig 5. Hidden latent features in LLMs

4.3 Hybrid Cluster

Hybrid works combine algorithmic innovation with empirical validation, edging toward grounded reasoning. DeepDive [1] and ARK-V1 [5] integrate knowledge graphs with reinforcement learning for interpretable search and question answering. HIRAG [7], Memory-R1 [8], and Fine-Tuning LLM Agents Without Fine-Tuning [11] use hierarchical reasoning and memory optimization to improve factuality and coherence. Seed Diffusion [12] and Dynamic Chunking [14] merge architectural design with measurable performance gains. These efforts bridge computation and observation, offering partial semantic grounding through retrieval, hierarchy, and context control.

4.4 Conceptual Map

Figure 6 illustrates the conceptual intersection between Algorithmic Correction, Empirical Pattern Analysis, and Semantic Grounding, unified through Linguistic Frames as the integrative layer. Algorithmic Correction represents the computational mechanisms that refine and align language models' outputs, while Empirical Pattern Analysis focuses on data-driven observations of linguistic behavior and usage patterns. Semantic Grounding, in turn, links linguistic symbols to real-world meaning and context. At their intersection, Linguistic Frames function as the conceptual bridge that connects structure, evidence, and meaning—facilitating coherence between algorithmic optimization, empirical regularities, and semantic interpretation. Together, these domains capture the multidimensional nature of grounded language understanding, where computational precision, observed data, and contextual meaning converge.

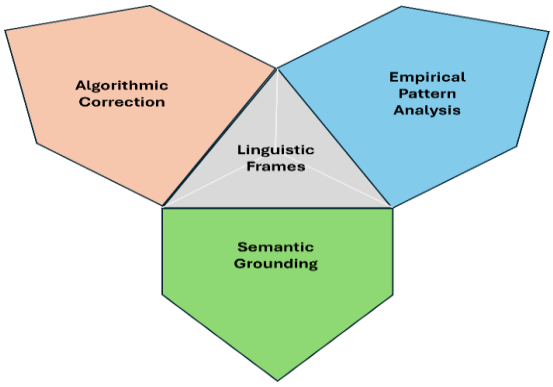

Fig 6. Conceptual Intersection between Algorithmic Correction, Empirical Pattern Analysis, and Semantic Grounding

Across all three clusters, a shared epistemic tension emerges: the disjunction between behavioral optimization and semantic representation. Algorithmic studies prioritize fluency and efficiency; empirical ones emphasize interpretability and user perception; hybrids attempt to reconcile both yet often lack explicit linguistic formalisms. What unites them is an absence of semantic architecture capable of constraining generative drift at its source. This comparative synthesis exposes several critical questions: Can semantic grounding be operationalized without restricting generative creativity? How might linguistic frames, as structured carriers of meaning, offer a scalable solution to this representational gap? The clustering system thus does more than categorize methodologies—it reveals the field's underlying architecture of thought, where the pursuit of truth in language models remains suspended between optimization, observation, and interpretation.

5. CONCLUSION

Hallucinations in large language models remain one of the most significant barriers to building trustworthy and reliable AI systems. While current mitigation strategies offer partial solutions, they often lack a sustainable mechanism for aligning generated outputs with truth and context. This paper has argued that linguistic frames provide a promising foundation for grounding by encoding structured meaning, roles, and relations directly into model architectures.

Recent research reveals a temporal shift—from reactive detection methods focused on post-hoc verification to proactive representational grounding that embeds semantic structure into the generation process itself. Within this evolution, a methodological lineage emerges. Algorithmic approaches like retrieval-augmented generation prioritize efficiency but often sacrifice semantic fidelity, while empirical studies emphasize contextual nuance yet lack architectural integration. Linguistic frame-based architectures synthesize these trajectories, bridging computational efficiency with semantic alignment.

By integrating these frames, generative systems can better anchor their outputs to coherent semantic structures, reducing factual drift and enhancing interpretability. The proposed architectural pathways—ranging from symbolic-neural hybrids to frame-augmented attention—illustrate how grounding can move beyond post-hoc corrections toward intrinsic design.

Future research should explore hybrid neural-symbolic architectures, multilingual scalability, and cross-domain adaptability, potentially visualized through a conceptual map that situates frame-grounded architectures within the broader landscape of hallucination mitigation. Ultimately, these clusters provide the structural context for examining how linguistic frames may serve as integrative semantic scaffolds, guiding the next generation of language models toward reliability, transparency, and human-aligned reasoning.

6. REFERENCES

[1] Lu, Rui, Zhenyu Hou, Zihan Wang, Hanchen Zhang, Xiao Liu, Yujiang Li, Shi Feng, Jie Tang, and Yuxiao Dong. "DeepDive: Advancing Deep Search Agents with Knowledge Graphs and Multi-Turn RL." arXiv preprint arXiv:2509.10446 (2025).

[2] Wiesner, Florian, Matthias Wessling, and Stephen Baek. "Towards a Physics Foundation Model." *arXiv preprint arXiv:2509.13805* (2025).

[3] Wang, Miles, Tom Dupré la Tour, Olivia Watkins, Alex Makelov, Ryan A. Chi, Samuel Miserendino, Johannes Heidecke, Tejal Patwardhan, and Dan Mossing. "Persona Features Control Emergent Misalignment." *arXiv preprint arXiv:2506.19823* (2025).

[4] Videau, Mathurin, Badr Youbi Idrissi, Alessandro Leite, Marc Schoenauer, Olivier Teytaud, and David Lopez-Paz.

"From Bytes to Ideas: Language Modeling with Autoregressive U-Nets." *arXiv preprint arXiv:2506.14761* (2025).

[5] Klein, Jan-Felix, and Lars Ohnemus. "ARK-V1: An LLM-Agent for Knowledge Graph Question Answering Requiring Commonsense Reasoning." *arXiv preprint arXiv:2509.18063* (2025).

[6] Bao, Wenrui, Zhiben Chen, Dan Xu, and Yuzhang Shang. "Learning to Parallel: Accelerating Diffusion Large Language Models via Adaptive Parallel Decoding." *arXiv preprint arXiv:2509.25188* (2025).

[7] Jiao, YiHan, ZheHao Tan, Dan Yang, DuoLin Sun, Jie Feng, Yue Shen, Jian Wang, and Peng Wei. "Hirag: Hierarchical-thought instruction-tuning retrieval-augmented generation." *arXiv preprint arXiv:2507.05714* (2025).

[8] Yan, Sikuan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Hinrich Schütze, Volker Tresp, and Yunpu Ma. "Memory-R1: Enhancing Large Language Model Agents to Manage and Utilize Memories via Reinforcement Learning." *arXiv preprint arXiv:2508.19828* (2025).

[9] Janiak, Denis, Jakub Binkowski, Albert Sawczyn, Bogdan Gabrys, Ravid Shwartz-Ziv, and Tomasz Kajdanowicz. "The Illusion of Progress: Re-evaluating Hallucination Detection in LLMs." *arXiv preprint arXiv:2508.08285* (2025).

[10] Hoang Nguyen, Minh, Van Dai Do, Dung Nguyen, Thin Nguyen, and Hung Le. "CausalPlan: Empowering Efficient LLM Multi-Agent Collaboration Through Causality-Driven Planning." *arXiv e-prints* (2025): arXiv-2508.

[11] Zhou, Huichi, Yihang Chen, Siyuan Guo, Xue Yan, Kin Hei Lee, Zihan Wang, Ka Yiu Lee et al. "Memento: Fine-tuning llm agents without fine-tuning llms." Preprint (2025).

[12] Song, Yuxuan, Zheng Zhang, Cheng Luo, Pengyang Gao, Fan Xia, Hao Luo, Zheng Li et al. "Seed diffusion: A large-scale diffusion language model with high-speed inference." arXiv preprint arXiv:2508.02193 (2025).

[13] Wang, Haozhe, Qixin Xu, Che Liu, Junhong Wu, Fangzhen Lin, and Wenhu Chen. "Emergent hierarchical reasoning in llms through reinforcement learning." *arXiv preprint arXiv:2509.03646* (2025).

[14] [14] Hwang, Sukjun, Brandon Wang, and Albert Gu. "Dynamic chunking for end-to-end hierarchical sequence modeling." arXiv preprint arXiv:2507.07955 (2025).

[15] Sheshadri, Abhay, John Hughes, Julian Michael, Alex Mallen, Arun Jose, and Fabien Roger. "Why Do Some Language Models Fake Alignment While Others Don't?." arXiv preprint arXiv:2506.18032 (2025).

[16] Yin, Fangcong, Xi Ye, and Greg Durrett. "Lofit: Localized fine-tuning on llm representations." *Advances in Neural Information Processing Systems* 37 (2024): 9474-9506.