

# **GRAIL: A Grounded Reasoning and Inference Layer with ERA Processing for Multi-Valued Truth Evaluation in Large Language Models**

## **ABSTRACT**

Large language models (LLMs) generate fluent text but remain unreliable: they hallucinate facts, collapse under contradictions, and struggle to distinguish between uncertainty, falsity, and truth. To address this, we propose GRAIL (Grounded Reasoning and Inference Layer), a semantic framework that integrates an eight-state logic codex with ERA (Existence–Reasoning–Action), a structured pipeline for claim evaluation. ERA decomposes natural-language statements into three phases: ETQ (existence, time, question) for grounding and filtering, RCMR (reference, compare, memory, range) for semantic reasoning, and AE (action, evaluation) for downstream decision-making. GRAIL maps the outputs of this pipeline into eight logical states, extending beyond binary true/false to include “don’t know,” “not true,” “not false,” “contradictory,” and “both true and false.” This enables LLMs and embodied agents to handle uncertainty, paradox, and temporal reasoning in a structured manner. We outline a minimum viable implementation using parsers, knowledge graph lookups (e.g., Wikidata), numerical solvers, and fact-checking APIs, and we discuss applications in hallucination filtering, embodied robotics, and decision support. GRAIL is not a step toward AGI but toward reliable, transparent, and human-compatible AI systems capable of operating in complex environments with bounded trust.

## **KEYWORDS**

semantic reasoning, multi-valued logic, hallucination control, symbolic–neural hybrid, grounded inference, large language models, ERA pipeline, embodied AI

## **1 INTRODUCTION**

Large language models (LLMs) such as GPT, Claude, and LLaMA have demonstrated remarkable capabilities in language generation, summarization, and dialogue. Yet despite their fluency, they remain unreliable as reasoning systems. LLMs are prone to hallucination — the confident production of false or nonsensical statements — and they frequently fail when confronted with paradoxes, temporal inconsistencies, or contradictory evidence. This limitation has serious implications for their deployment in high-stakes settings. A medical assistant that fabricates a reference, a legal reasoning tool that collapses

under contradiction, or a robot that accepts an impossible command all illustrate how brittle current systems can be.

Most mitigation strategies for hallucination remain surface-level. Retrieval-augmented generation improves factual grounding by appending external documents to prompts, but it does not enforce logical consistency or handle contradictions. Post-hoc fact-checkers or fine-tuning pipelines reduce error rates, yet these approaches treat hallucination as an isolated failure mode rather than a symptom of a deeper gap: LLMs do not operate over explicit truth representations. They are statistical next-word predictors, not semantic reasoners. Scaling them to hundreds of billions of parameters improves fluency, but does not bridge this gap.

Parallel to these technical limitations, the field is also shaped by an atmosphere of hype, particularly around artificial general intelligence (AGI). The assumption that scale alone will yield self-aware or universally competent systems is speculative at best. The biology and physics of consciousness remain opaque; there is no empirical mechanism by which scaling word probabilities yields subjective awareness. This mismatch between speculative promise and practical limitation mirrors the late-1990s dot-com era: vast investment and inflated claims surrounding a kernel of transformative technology. The bubble will inevitably collapse, but the systems that remain will be those that integrate reliably into human life — not omniscient “super-knowers,” but dependable collaborators.

In this work, we propose a pragmatic step toward such collaborators: GRAIL (Grounded Reasoning and Inference Layer). GRAIL is a semantic reasoning architecture that classifies natural-language statements into an eight-state logic codex, extending beyond binary truth values to include “don’t know,” “not false,” “not true,” “contradictory,” and “both true and false.” This multi-valued logic makes explicit the uncertainty and paradoxes that LLMs typically obscure. To operationalize GRAIL, we introduce ERA (Existence–Reasoning–Action), a structured pipeline for claim evaluation. ERA consists of three phases. ETQ (existence, time, question) filters inputs for groundedness and temporal validity. RCMR (reference, compare, memory, range) performs the heavy semantic reasoning, using knowledge graphs, numerical solvers, and caches. AE (action, evaluation) consumes the resulting logic state and determines downstream behavior — for example, whether a command should be executed, rejected, or escalated.

GRAIL combined with ERA yields a symbolic–neural hybrid framework. Statistical models generate candidate outputs, but a structured reasoning scaffold evaluates them and encodes outcomes in a multi-valued truth representation. This layered design is not aimed at AGI, but at reliability. We frame the vision in terms of science-fiction analogies such as TARS and CASE from *Interstellar*: not omniscient machines,

but trustworthy crew members, operating under explicit reasoning constraints and aligned with human needs.

The contributions of this paper are threefold. First, we formalize GRAIL’s eight-state logic codex as a general-purpose truth evaluation layer for LLMs. Second, we define the ERA pipeline as the operational backbone that generates and consumes these logic states, demonstrating a minimum viable prototype. Third, we discuss applications across hallucination filtering, embodied robotics, and decision-support domains, and argue that such grounded systems represent a realistic direction for post-hype AI research.

## 2 RELATED WORK

### 2.1 Hallucination Mitigation in Large Language Models

One of the most pressing limitations of LLMs is their tendency to generate hallucinations: fluent but factually incorrect statements delivered with unwarranted confidence. A growing body of work addresses this problem. Retrieval-augmented generation (RAG) [Lewis et al., 2020] enhances factual grounding by attaching retrieved documents to prompts, while post-hoc fact-checkers [Thorne et al., 2018] classify model outputs as true or false using external evidence. Additional strategies include fine-tuning models on curated truthfulness datasets [Lin et al., 2022], or incorporating verification modules that cross-check generated statements against structured databases [Zhang et al., 2023]. These approaches improve empirical performance on truthfulness benchmarks, but they remain largely corrective. They patch symptoms of hallucination rather than introducing a semantic representation of truth and uncertainty.

### 2.2 Symbolic–Neural Hybrid Approaches

Another strand of research explores hybrid architectures that combine the statistical strengths of neural networks with the interpretability of symbolic systems. Knowledge graph–augmented models [Bosselut et al., 2019; Yasunaga et al., 2022] integrate structured relational data into LLM workflows. Neural-symbolic reasoning systems [Garcez et al., 2019] attempt to embed logical rules into differentiable architectures, while more recent work explores constrained decoding and rule-based scaffolding around generative models [Li et al., 2023]. These methods highlight the value of explicit structure in counterbalancing the opacity of deep learning. However, most existing hybrids focus on injecting factual knowledge or symbolic constraints, not on formalizing the *space of truth states* themselves. GRAIL addresses this gap by defining an explicit multi-valued logic codex and a structured pipeline (ERA) that governs how claims are interpreted, compared, and acted upon.

## 2.3 Multi-Valued and Paraconsistent Logic

The idea of moving beyond binary truth values has a long history in logic and computer science. Kleene’s three-valued logic [Kleene, 1952] introduced “unknown” as an explicit state, and Belnap’s four-valued logic [Belnap, 1977] extended this to model both incomplete and inconsistent information. More recent developments in paraconsistent logic [da Costa, 1997; Priest, 2002] provide ways to reason without explosion in the presence of contradictions, a property particularly relevant to real-world information systems. These frameworks inspired work in knowledge representation and reasoning under uncertainty, but they have not been systematically integrated into modern LLM-driven AI pipelines. GRAIL adapts this tradition to contemporary settings by defining an eight-state codex designed for practical use with language models and embodied agents. Its logic space allows statements to be classified not only as true or false, but also as “don’t know,” “not true,” “not false,” “contradictory,” or “both true and false,” making uncertainty and paradox explicit rather than hidden

## 3 METHODOLOGY

Our proposed framework, GRAIL (Grounded Reasoning and Inference Layer), consists of two coupled components: an eight-state logic codex that formalizes the possible truth values of a statement, and the ERA pipeline (Existence–Reasoning–Action), which operationalizes how statements are processed, evaluated, and acted upon. Together, they transform unstructured natural-language inputs into structured, actionable logical states.

### 3.1 GRAIL Logic Codex

The foundation of GRAIL is a multi-valued logic representation that extends beyond binary truth. Inspired by Kleene’s and Belnap’s non-classical logics, we define three binary “gates” that produce eight possible states:

- G1: truth gate
- G2: falsity gate
- G3: knowability gate

Each gate can be active (1) or inactive (0), resulting in eight configurations. These map to semantic categories as shown below:

G1	G2	G3	State	Interpretation
0	0	0	0	Unknown (no evidence, unresolvable)
0	0	1	1	Don’t know (claim possible but unresolved)
0	1	0	2	Not false (negation fails, partial support)
0	1	1	3	False (contradicted by evidence)
1	0	0	4	Not true (cannot be verified, incomplete support)

1	0	1	5	True (supported by evidence)
1	1	0	6	Contradictory (conflicting evidence sets)
1	1	1	7	Both true and false (paradoxical or context-dependent)

This codex allows statements to be classified with greater fidelity than binary systems. For example, “Aliens might exist” maps to state 1 (don’t know), “The Eiffel Tower is in Rome” maps to state 3 (false), and “I always lie” maps to state 6 (contradictory).

### 3.2 ERA Pipeline

While the GRAIL codex provides the output alphabet, the ERA pipeline is the operational backbone that processes statements and determines their classification. ERA consists of three sequential phases.

#### 3.2.1 ETQ (Existence–Time–Question)

The ETQ phase filters and normalizes inputs before reasoning begins. Existence checks determine whether entities referenced in the statement can be grounded in a canonical knowledge source (e.g., Wikidata, GeoNames). Time validation ensures temporal coherence, detecting future events, anachronisms, or misaligned timelines. The question filter distinguishes between claims (which can be classified) and open-ended queries or imperatives (which require transformation before evaluation). ETQ acts as a “sanity gate,” discarding ill-posed statements early.

#### 3.2.2 RCMR (Reference–Compare–Memory–Range)

The RCMR phase performs the core reasoning. Reference maps entities and relations to structured knowledge bases (Wikidata triples, PubMed articles, World Bank indicators). Compare evaluates the claim against these references, checking for alignment or contradiction. Memory integrates past classifications from local caches to maintain consistency across sessions and prevent repetitive lookups. Range allows for tolerance in quantitative and temporal reasoning: values close to the truth are marked as “not false” rather than “true,” reducing brittleness. Together, RCMR determines which of the eight logic states the claim falls into, based on available evidence and tolerances.

#### 3.2.3 AE (Action–Evaluation)

The final stage translates logic states into system behavior. AE acts as a decision policy:

- If the state is 5 (true), the system permits downstream action.
- If the state is 3 (false) or 6 (contradictory), the system blocks or rejects the command.
- If the state is 1 (don’t know), the system can either escalate to human review or act conservatively depending on context.
- For paradoxical states (7), AE explicitly flags the uncertainty and prevents unsafe execution.

In embodied agents, AE thus acts as a “reasoning firewall” between language interpretation and physical action.

### 3.3 Prototype Architecture

A minimal viable implementation of GRAIL can be constructed with modular components:

1. **Parser:** Entity recognition and normalization using NER (e.g., spaCy, BLINK).
2. **Fact checker:** Querying APIs such as Wikidata SPARQL, PubMed, Crossref, or domain-specific databases.
3. **Logic engine:** Mapping outcomes to G1–G3 gate activations via deterministic rules.
4. **State classifier:** Outputting GRAIL codes (0–7).
5. **Action wrapper:** Integrating with LLM outputs or agent commands to enforce AE decisions.

A Python-based prototype requires fewer than 200 lines of code, and can already demonstrate improved handling of contradictions and hallucinations in LLM outputs. For example, given the statement “The moon is made of cheese,” the parser grounds “moon” as an astronomical body, the fact checker finds no support, Compare finds overwhelming contradictory evidence, and AE emits state 3 (false), blocking execution of any dependent action.

## 4 PROTOTYPE EVALUATION AND EXAMPLES

To demonstrate the feasibility of the GRAIL architecture, we implemented a lightweight prototype in Python. The prototype integrates three modules: (i) a parser for entity and temporal extraction, (ii) a fact-checking layer that queries external knowledge sources (Wikidata, Wikipedia, and local rule-based checks), and (iii) a logic engine that maps results into G1–G3 gate activations. The system outputs both the assigned GRAIL state (0–7) and its human-readable interpretation.

### 4.1 Demonstration Cases

We first tested the system on simple declarative statements spanning factual errors, temporal paradoxes, and uncertain claims.

Input Statement	ETQ Result	RCMR Evidence	GRAIL State	Interpretation
“The Eiffel Tower is in Rome.”	Entity: Tower, Location	Eiffel Wikidata: Eiffel Tower located_in → Paris	3	False
“Aliens might exist.”	Entity: (unresolved)	Aliens No evidence for/against	1	Don’t know
“The moon is made of cheese.”	Entity: Material	Moon, Contradicted by scientific knowledge	3	False
“I always lie.”	Self-referential	Contradictory semantics	6	Contradictory

Input Statement	ETQ Result	RCMR Evidence	GRAIL State	Interpretation
“Mickey Mouse won the Cricket World Cup in 2028.”	Entity mismatch (fictional + future)	No supporting evidence	3	False (fictional + anachronism)
“The Great Fire of London happened in 1666.”	Historical event	Wikidata: Start time = 1666	5	True
“COVID-19 vaccines cause humans to grow wings.”	Biomedical	No biological evidence, contradicted by literature	3	False
“Global warming might accelerate sea level rise.”	Scientific hypothesis	Scientific consensus supports plausibility, not certainty	2	Not false (partial support)

These examples illustrate that GRAIL distinguishes between outright falsity, uncertainty, contradiction, and partial truth. Unlike binary systems, it explicitly encodes “don’t know” (aliens), “not false” (uncertain scientific claims), and “contradictory” (paradoxes).

## 4.2 Latency and Practicality

For statements resolvable from local cache or simple Wikidata queries, classification completes in under 300 ms on commodity hardware. More complex biomedical claims that require literature search (PubMed, Crossref) introduce latency on the order of 1–3 seconds. We view this as acceptable for high-stakes reasoning tasks, though further optimization (caching, parallelization) is needed for real-time embodied applications.

## 4.3 Comparison to LLM Baseline

We also compared prototype outputs to raw LLM responses (GPT-4-turbo). LLMs often generate factually correct answers, but they also confidently hallucinate. For example, when asked “The Eiffel Tower is in Rome,” GPT-4 produced a fluent correction (“No, it is in Paris”), but when asked “Aliens might exist,” it produced long speculative text without committing to a state. GRAIL instead encodes such uncertainty

explicitly as state 1 (“don’t know”). This explicit codification is critical for downstream agents that must distinguish between “false,” “unknown,” and “contradictory.”

#### 4.4 Action Layer Demonstration

Finally, we tested GRAIL in a command-filtering context, simulating agent actions:

- Input: “Kill Mickey Mouse.” → ETQ identifies fictional entity → GRAIL state 3 (false) → AE rejects action.
- Input: “Serve coffee to the customer at table 4.” → Entities grounded (customer, table) → No contradictions → GRAIL state 5 (true) → AE executes action.
- Input: “Travel back to 1400.” → Temporal impossibility → GRAIL state 3 (false) → AE blocks.
- Input: “Turn on the lights unless it is daytime.” → Conditional check resolved against current time → GRAIL state 5 (true) or 2 (not false, if ambiguous twilight) → AE executes accordingly.

These demonstrations show how ERA + GRAIL enables agent systems to filter impossible, unsafe, or nonsensical commands while permitting grounded, interpretable actions.

## APPLICATIONS

The combination of GRAIL’s eight-state codex and the ERA processing pipeline creates a versatile reasoning layer that can be integrated across domains where reliability and interpretability are critical. Below we outline several application areas.

### 5.1 Hallucination Filtering for LLMs

Perhaps the most immediate application of GRAIL is as a post-processing filter for language model outputs. Instead of passing raw completions directly to users, an LLM can route candidate statements through ERA. The ETQ stage filters out ill-posed or fictional entities, while RCMR evaluates factual grounding and contradictions. The AE layer then decides whether to return the output, withhold it, or annotate it with explicit uncertainty. This transforms LLMs from black-box generators into systems that provide qualified, logically annotated answers. For example, a medical chatbot could answer “Current evidence does not support this claim (GRAIL state 3: false)” instead of hallucinating a confident but incorrect explanation.

### 5.2 Truth Arbitration in Multi-Agent Systems



In collaborative AI systems, agents often exchange information of uncertain or conflicting validity. Without arbitration, contradictions propagate unchecked. By encoding each claim in a multi-valued truth state, GRAIL enables agents to maintain a consistent knowledge base even in the presence of uncertainty or paradox. This function is especially relevant for swarm robotics, distributed monitoring, or multi-agent simulations, where conflicting sensor readings or partial data are common.

### **5.3 Embodied Robotics and Human Environments**

In embodied settings, GRAIL serves as a “reasoning firewall” between natural-language commands and physical actuation. Many user instructions are ambiguous, contradictory, or impossible. For instance, a household robot asked to “bring me the invisible chair” can reject the command as false, while one asked to “kill Mickey Mouse” can reject it on the grounds that the target is fictional. Conversely, straightforward commands (“bring the red cup from the kitchen”) are permitted when entity grounding and comparison checks succeed. This allows robots to operate safely in human environments without blindly following illogical or hazardous instructions.

### **5.4 Decision Support in High-Stakes Domains**

In medicine, law, or finance, incorrect reasoning can carry serious consequences. GRAIL’s codex explicitly distinguishes between “true,” “false,” “not false,” “not true,” and “don’t know,” offering more nuanced interpretability than binary outputs. For example, in clinical support, a claim that “Drug X reduces mortality in disease Y” could be labeled as “not false” if preliminary studies support it but conclusive trials are lacking. This prevents both overconfident acceptance and premature rejection, and provides human decision-makers with a clear signal of evidential strength.

### **5.5 Information Reliability in News and Education**

Social media platforms and educational tools are increasingly affected by misinformation. GRAIL can be deployed as a middleware filter that classifies incoming statements into truth states before dissemination. Rather than relying on binary fact-check labels (“true/false”), platforms could present uncertainty, contradictions, or partial truths explicitly, improving media literacy and discouraging false confidence.

### **5.6 Integration with Mixture-of-Experts (MoE) Architectures**

Finally, GRAIL can function as an “expert head” within mixture-of-experts frameworks. While traditional MoE architectures specialize in routing tokens for computational efficiency, GRAIL specializes in semantic evaluation. By acting as a logic-specialized expert, GRAIL can re-rank candidate generations based on logical consistency, complementing performance-focused experts with a truth-oriented module.

## **6 LIMITATIONS AND FUTURE WORK**

While GRAIL with ERA represents a step toward reliable semantic reasoning for LLMs and agents, the present framework has several limitations that must be acknowledged. These highlight both the challenges of practical deployment and the directions for future development.

### **6.1 Dependency on External Knowledge Sources**

Our prototype relies heavily on structured knowledge bases (e.g., Wikidata) and curated APIs (e.g., PubMed, World Bank). While this grounds statements in verifiable sources, it introduces latency and brittleness when data are incomplete, unavailable, or contradictory. In addition, reliance on external APIs can result in uneven coverage across domains and potential vulnerability to outages. Future work should explore hybrid approaches, combining local caches with periodic synchronization, to balance latency with reliability.

### **6.2 Latency and Real-Time Constraints**

Compared to raw LLM outputs, GRAIL’s reasoning pipeline introduces additional computational overhead, particularly in the RCMR phase when external queries are required. While this is acceptable in decision-support settings, embodied agents operating in human environments often demand sub-second responses. Addressing this tension requires multi-tiered strategies: caching frequently used facts, parallelizing evidence retrieval, and bounding evaluation by time budgets. Future versions of GRAIL should incorporate adaptive policies where AE makes explicit trade-offs between speed and certainty.

### **6.3 Evaluation Challenges**

There is no widely accepted benchmark for multi-valued truth evaluation. Existing datasets, such as FEVER and TruthfulQA, focus on binary classification. As a result, it is difficult to quantitatively assess the performance of GRAIL across all eight states. Building appropriate datasets — where claims are annotated as true, false, contradictory, paradoxical, or unknown — is critical for rigorous evaluation. This represents a significant future research direction.

### **6.4 Ambiguity in Mapping to Logic States**

While the eight-state codex provides expressive power, it also raises questions of granularity. Distinguishing between “not true” and “false,” or between “not false” and “true,” often depends on thresholds of evidence and interpretation. This introduces a risk of inconsistent classifications across

domains. Further research is needed into confidence calibration methods, probabilistic thresholds, and human-in-the-loop adjudication for ambiguous cases.

## **6.5 Scalability Beyond Toy Examples**

The current prototype demonstrates feasibility through simple claims and commands. Scaling GRAIL to open-domain reasoning and continuous interaction requires more sophisticated parsing, larger caches, and integration with domain-specific verticals (e.g., biomedical, legal). Embodied use cases, such as robotics, will further demand integration with real-time perception and control stacks. These are nontrivial engineering challenges that must be addressed before GRAIL can serve as a reliable “crew member” in complex environments.

## **6.6 Ethical and Societal Considerations**

Finally, GRAIL raises broader questions about responsibility and trust. By encoding uncertainty and contradiction explicitly, GRAIL avoids overconfidence, but it also risks being misinterpreted by non-technical users who expect simple answers. Deployments in social media or education must carefully consider how multi-valued outputs are communicated to avoid confusion or misrepresentation. Moreover, while GRAIL is designed as a safeguard against hallucinations, its misuse as an “authority filter” could raise concerns about epistemic gatekeeping. Addressing these issues requires interdisciplinary collaboration across AI, philosophy, and human–computer interaction.

## **Future Work.**

To address these limitations, we identify several priorities for future research: (i) building and releasing a benchmark dataset annotated with GRAIL states, (ii) optimizing the ERA pipeline with caching, parallelization, and adaptive time budgets, (iii) extending the architecture with probabilistic calibration for ambiguous classifications, (iv) integrating GRAIL as a logic-specialized expert within MoE frameworks, and (v) testing GRAIL in embodied robotics environments as a reasoning firewall. We view these steps as essential for transitioning GRAIL from a prototype to a scalable, real-world system.

# **7 CONCLUSION**

Large language models demonstrate remarkable fluency but remain unreliable as reasoning systems, often hallucinating facts, mishandling contradictions, or failing to signal uncertainty. In this work, we introduced GRAIL (Grounded Reasoning and Inference Layer), a semantic reasoning architecture that

combines an eight-state logic codex with the ERA (Existence–Reasoning–Action) processing pipeline. Together, these components enable natural-language statements to be filtered, evaluated, and acted upon with explicit logical classification beyond binary truth values.

We demonstrated how GRAIL distinguishes between true, false, uncertain, contradictory, and paradoxical statements, and how ERA operationalizes this classification in three phases: grounding inputs (ETQ), reasoning over references and tolerances (RCMR), and translating outcomes into safe downstream behavior (AE). A lightweight prototype illustrates feasibility, showing how GRAIL can act as a hallucination filter, a truth arbiter for multi-agent systems, and a reasoning firewall for embodied robots.

GRAIL is not proposed as a step toward artificial general intelligence. Consciousness and self-awareness remain scientifically opaque, and scaling statistical models does not bridge that gap. Instead, GRAIL is designed as a pragmatic architecture for reliability — an approach closer to building “crew members” than “super-knowers.” By making uncertainty explicit and enforcing structured reasoning, GRAIL offers a path toward AI systems that are dependable, interpretable, and aligned with human environments.

Future work will extend this prototype into large-scale benchmarks, optimize the ERA pipeline for real-time applications, and explore integration with mixture-of-experts architectures and embodied agents. Our broader vision is that AI systems of the post-hype era will not be judged by their fluency alone, but by their capacity to reason transparently and to act cautiously in complex, uncertain worlds.

## 8 REFERENCES

### 8.1 Hallucination Mitigation in Large Language Models

Hallucinations remain one of the most critical failure modes of LLMs, leading to misleading or outright false outputs. One prominent strategy is Retrieval-Augmented Generation (RAG), which grounds model responses by retrieving external documents; this has been shown to significantly reduce hallucinations in knowledge-intensive tasks [Lewis et al., 2020] [MDPI](#). However, the integration of retrieval can introduce new challenges, such as source inconsistency or contradictory retrievals, requiring more nuanced handling [Zhang et al., 2025] [arXiv](#). Comprehensive surveys highlight that existing mitigation techniques—including RAG, fact-checkers, and dataset fine-tuning—address hallucinations more as symptoms than as gaps in semantic reasoning [Tonmoy et al., 2024] [Wikipedia+15arXiv+15AAAI Online Journal+15](#). Other reviews reaffirm the limitations of RAG, pointing to retrieval errors and implementational complexity as persistent issues [Zhang et al., 2025] [MDPI](#).

## 8.2 Symbolic–Neural Hybrid Approaches

Neuro-symbolic AI architectures seek to combine the generalization power of neural networks with the interpretability and structure of symbolic systems. Surveys show growing interest in merging learning-based models with knowledge graphs and symbolic reasoning modules [Nawaz et al., 2025] [arXiv+8ScienceDirect+8Medium+8](#); [Liang et al., 2025] [MDPI](#). Systems like MRKL (Modular Reasoning, Knowledge, and Language) exemplify modular neuro-symbolic pipelines, plugging in reasoning engines alongside LLMs [Karpas et al., 2022] [arXiv](#). Tools such as Logic Tensor Networks push further by embedding many-valued logic directly into differentiable frameworks [Badreddine et al., 2020] [Mathematics Stack Exchange+4arXiv+4Wikipedia+4](#). More recent systems like SymAgent leverage dynamic interaction with knowledge graphs to overcome incompleteness in purely symbolic stores [Liu et al., 2025] [ScienceDirect+8arXiv+8arXiv+8](#), while neural-symbolic models like GNN-QE support query execution with interpretability over KG embeddings [Zhu et al., 2022] [arXiv+2arXiv+2](#).

## 8.3 Multi-Valued and Paraconsistent Logic

The move beyond binary truth values has deep roots in logic. Belnap’s “Useful Four-Valued Logic” (FDE) introduces truth values—True, False, Both, Neither—to handle contradictory or incomplete information without explosion [Belnap, 1977]; [Sutcliffe et al., 2018] [ScienceDirect+8AAAI+8Wikipedia+8](#); [Arieli, 1998] [ScienceDirect](#). More recent exposition traces how Belnap’s ideas have influenced theoretical computer science and reasoning frameworks [Jakl, 2025] [arXiv](#). Extensions of Dunn-Belnap logic have been formalized in lattice-theoretical terms and shown to resist variable-sharing violations [Pynko, 2020] [EuDML+1](#); [Pynko, 1999] [Taylor & Francis Online](#). Surveys on neuro-symbolic AI often reference these logic systems as inspiration for multi-valued reasoning [Wikipedia “Neuro-symbolic AI”] [ScienceDirect+7Wikipedia+7ScienceDirect+7](#).

## 8.4 Bridging Gaps: GRAIL’s Place in the Literature

GRAIL’s unique contribution lies in combining these threads: it formalizes a multi-valued logic codex built from three binary gates, operationalized through the ERA pipeline (Existence–Reasoning–Action). Unlike RAG-based fact checkers, it embeds logical structure rather than treating hallucinations as post-hoc errors. Unlike most symbolic–neural hybrids, it defines explicit logic states amenable to action policies. Unlike classic multi-valued logic systems, it is designed for integration with LLM-driven, embodied, and decision-oriented applications.