Beyond Retrieval: Bidirectional Neuro-Symbolic Validation for Reliable Graph-RAG

Leonardo Arazo de Oliveira Araújo, Didier Vega-Oliveros

Institute of Science and Technology, Federal University of São Paulo (UNIFESP), São José dos Campos, SP, Brazil
arazo.leonardo@unifesp.br, didier.vega@unifesp.br

Abstract. Retrieval-Augmented Generation (RAG) and its graph-based extensions improve factual grounding in LLMs but do not guarantee logical consistency when responses must satisfy explicit domain constraints. We propose a bidirectional neuro-symbolic validation framework for Graph-RAG that combines First-Order Logic reasoning over a domain knowledge graph with an Annotated Datalog engine built on PyReason, propagating confidence bounds across inferred facts. A bidirectional mechanism supports symbolic enrichment, post-hoc validation with corrective rewriting, and direct symbolic execution. In a Brazilian public university advisory benchmark, the framework achieves strict accuracy of 84.2% and weighted accuracy of 87.7% (with partial-credit weighting), an 87% reduction in factual errors relative to LLM-only; a backend-sensitivity study with `gemma4:12b` preserves the baseline ordering with no answer labeled incorrect (0/57).

CCS Concepts: • **Computing methodologies** → **Knowledge representation and reasoning**; • **Information systems** → *Retrieval models and ranking*.

Keywords: graph-rag, knowledge graphs, neuro-symbolic reasoning, retrieval-augmented generation

1. INTRODUCTION

Large Language Models (LLMs) have become an effective interface for accessing structured knowledge through natural language. Retrieval-Augmented Generation (RAG) improves factual grounding by incorporating external evidence into the generation process [Cheng et al. 2025; Brown et al. 2025]. At the same time, Graph-RAG extends this paradigm by leveraging knowledge graphs to support relational and multi-hop retrieval [Zhang et al. 2025; Zhu et al. 2025]. Despite these advances, retrieval alone does not guarantee logical consistency: generated answers may remain grounded in retrieved evidence while violating domain-specific constraints [Han et al. 2025] that emerge from composing multiple facts. This limitation is particularly relevant in domains whose knowledge is naturally represented as dependency graphs, such as educational curricula, workflow planning, software dependencies, and regulatory compliance. In these settings, correctness depends not only on retrieving relevant information but also on reasoning over chains of dependencies and validating conclusions against explicit rules. In particular, Neuro-symbolic AI offers a promising approach to address this challenge [Garcez and Lamb 2023]. Still, existing methods typically employ symbolic reasoning either inside specialized learning architectures or as isolated post-processing checks [Yu et al. 2023; Saha et al. 2026], limiting the interaction between neural and symbolic components in practical Graph-RAG systems.

We propose a bidirectional neuro-symbolic validation framework that combines First-Order Logic (FOL) reasoning over a domain Knowledge Graph (KG) with an Annotated Datalog engine implemented through PyReason. The framework enables symbolic enrichment before generation, post-generation validation with corrective rewriting, and direct symbolic execution for compositional queries, while supporting confidence-aware reasoning by propagating probabilistic bounds over inferred rela-

Copyright©2026 Permission to copy without fee all or part of the material printed in KDMiLe is granted provided that the copies are not made or distributed for commercial advantage, and that notice is given that copying is by permission of the Sociedade Brasileira de Computação.

tions. We evaluate the framework on a university advisory benchmark derived from institutional data at a Brazilian public university. The main contributions are a bidirectional neuro-symbolic framework for Graph-RAG that integrates symbolic enrichment, post-generation validation, direct symbolic execution, and confidence-aware reasoning through Annotated Datalog, and an experimental evaluation on a real-world academic-advisory benchmark demonstrating substantial reductions in factual errors compared with LLM-only, RAG, and Graph-RAG baselines.

2. RELATED WORK

Retrieval-Augmented Generation (RAG) improves factual grounding by conditioning LLM outputs on external evidence [Cheng et al. 2025; Brown et al. 2025], but text-based retrieval often struggles with queries that require reasoning across multiple relational constraints. Graph-RAG addresses this limitation through graph-structured representations and multi-hop traversal [Zhang et al. 2025], including knowledge-graph-guided retrieval [Zhu et al. 2025] and graph-grounded question decomposition [Linders and Tomczak 2025]. Recent studies report consistent improvements on relation-intensive tasks [Han et al. 2025] and high-stakes domains such as medicine [Wu et al. 2024]. Nevertheless, graph retrieval provides supporting evidence rather than logical verification, leaving generated responses vulnerable to violations of domain-specific constraints.

On the other hand, Neuro-symbolic AI combines the pattern-recognition capabilities of neural models with the explicit reasoning mechanisms of symbolic systems [Garcez and Lamb 2023]. Existing approaches range from tightly integrated architectures with differentiable reasoning [Yu et al. 2023] to loosely coupled pipelines that verify or refine generated outputs, with recent work showing improvements in QA through KG-grounded symbolic execution [Saha et al. 2026; Saxena and Gaur 2026]. Most existing systems employ symbolic reasoning as a single-stage process, either before or after generation. In contrast to these single-stage designs, our framework couples the two directions in one loop: symbolic facts condition generation, and generation is then re-validated symbolically with corrective rewriting, and adds direct symbolic execution for compositional queries, a combination that, to our knowledge, prior Graph-RAG validation pipelines do not offer. Because real-world KGs are incomplete and noisy, we use the Annotated Datalog framework PyReason [Aditya et al. 2023] to propagate confidence bounds over inferred relations, allowing the same validation pipeline to operate on both curated and heuristically extracted knowledge.

3. BIDIRECTIONAL NEURO-SYMBOLIC FRAMEWORK

Hence, we propose a bidirectional neuro-symbolic framework for Graph-RAG, as illustrated in Figure 1. The architecture integrates four main components: a hybrid retrieval over the document corpus, a domain KG built from the same sources, a *symbolic reasoning layer* that supports FOL inference with probabilistic bounds, and a *bidirectional validation loop* that couples the symbolic layer with neural generation. An intent classifier and multi-agent router dispatch each query to the appropriate combination of components. First, the institutional documents (course descriptions, regulations, faculty profiles) are parsed into semantically structured chunks with section-level metadata, enabling intent-aware retrieval. We combine dense retrieval with Maximum Marginal Relevance (MMR) over vector embeddings for semantic similarity, and BM25 for exact lexical matching. The two rankings are fused through Reciprocal Rank Fusion (RRF) $\quad \mathrm{RRF_score}(d) = \sum_i \frac{1}{k + r_i(d)}, \quad$ where $r_i(d)$ is the rank of document d under retriever i and $k = 60$ is the standard rank-smoothing constant. The top-N list feeds the downstream Graph-RAG pipeline; RRF avoids score calibration across heterogeneous retrievers and is robust on both semantic and keyword-oriented queries.

The KG was implemented as a `MultiDiGraph` in NetworkX and built from the same markdown sources used for retrieval (Figure 2). It captures academic entities, i.e., disciplines, faculty members,

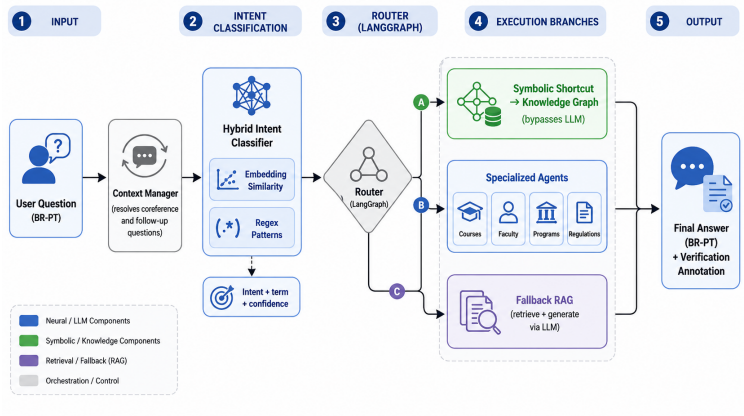

Fig. 1. Proposed bidirectional neuro-symbolic Graph-RAG framework based on hybrid retrieval, knowledge-graph reasoning, and symbolic validation.

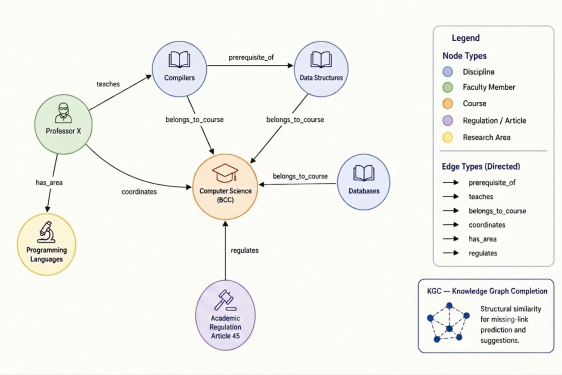

Fig. 2. Schema of the academic knowledge graph, showing main entity and relation types.

Table I. First-order logic rules implemented by the symbolic reasoning layer.

Rule	Formalisation	Semantics
R1	$\text{prereq}(x,y) \wedge \text{prereq}(y,z) \rightarrow \text{prereq_trans}(x,z)$	Transitive closure of dependency
R2	$(\forall p \in \text{prereqs}(d) : p \in C) \rightarrow \text{unlocked}(d,C)$	Discipline unlocked given completed set C
R3	$\lvert\text{dependents}(x)\rvert \geq \theta \rightarrow \text{critical}(x),\ \theta = 3$	Critical-node detection
R4	$\text{prereq}(x,z) \wedge \text{prereq}(y,z), x \neq y \rightarrow \text{co_prereq}(x,y)$	Shared downstream dependency
R5	$\min\{P : \text{taken}(P) \rightarrow \text{unlocked}(t, C \cup P)\}$	Minimal study-plan via topological BFS

programs, regulations, research areas, together with the relations that connect them. Edges of type `prerequisite_of` carry a confidence $c \in [0,1]$ distinguishing *curated* relations from the official curriculum ($c = 1.0$) from *heuristic* relations inferred from textual cues ($c < 1.0$); these values feed the probabilistic bounds propagated by the symbolic layer. The KG maintains accent-insensitive indices over names, codes, and aliases (via NFD normalization) to support reliable symbolic lookup despite heterogeneous naming.

The symbolic layer combines five FOL rules with graph-based procedures executed by an *Annotated Datalog* engine built on PyReason [Aditya et al. 2023]. Together, these rules capture the compositional patterns required to reason over a directed acyclic graph (DAG) of academic dependencies (Table I). Rule R1 is implemented directly as an Annotated Datalog program; R2-R5 operate over the inferred dependency structure produced by the symbolic layer.

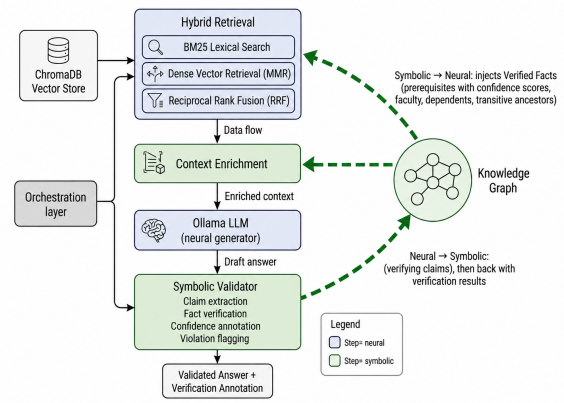

Fig. 3. Per-agent bidirectional validation loop. The symbolic layer provides pre-generation KG fact injection, post-generation output validation, and direct symbolic shortcuts for compositional intents.

Probabilistic bounds via max-bottleneck path. Under Annotated Datalog, each atom is associated with a confidence interval $[l, u] \subseteq [0, 1]$. Curated prerequisite edges are assigned $[1.0, 1.0]$, heuristic edges $[c, 1.0]$ with $c < 1$. For derived atoms, we adopt a *max-bottleneck path* semantics: among alternative derivations of prereq_trans(x, z), the system selects the path whose minimum edge confidence is maximal. For example, the chain $A \rightarrow_{1.0} B \rightarrow_{0.7} C \rightarrow_{1.0} D$ yields prereq_trans$(A, D) = [0.7, 1.0]$; if an alternative fully curated path exists, it dominates and produces $[1.0, 1.0]$. This reflects the user-facing intent: when multiple chains support a derivation, report the strongest one.

The bidirectional validation loop (Figure 3) establishes a two-way interaction between neural generation and symbolic reasoning and operates in three complementary modes. **Symbolic → Neural (pre-generation enrichment).** For selected intents, a block of KG-verified facts, direct prerequisites, transitive chains with confidence bounds, dependents, faculty assignments, and outputs of R1-R5, is prepended to the agent prompt before generation, reducing the surface from which unsupported claims can arise.

Neural → Symbolic (post-generation validation with corrective rewriting). Factual claims are extracted from the LLM output through pattern matching over recognized entities and relations, and are verified against the KG. When a violation is detected (e.g., a prerequisite not supported by the dependency closure), the framework performs a *corrective rewrite*: it replaces the response with an answer derived directly from the KG and the applicable rules. Inferred relations whose confidence is below 1.0 are surfaced as user-facing indicators (e.g., "probable" for $[0.7, 1.0]$).

Direct symbolic shortcut. For strictly compositional intents, program listings, curriculum exploration, prerequisite-chain analysis, course-faculty assignments, the framework bypasses LLM generation. It returns an answer derived directly from the KG, eliminating hallucination risk and reducing latency. Answer synthesis uses Ollama with `ministral-3:8b` under low-temperature decoding; prompts constrain the model to the provided context and instruct it to explicitly acknowledge missing information.

4. METHODOLOGY AND EXPERIMENTAL RESULTS

The evaluation was guided by four questions: **RQ1.** Does the framework improve factual consistency over LLM-only, RAG, and Graph-RAG baselines? **RQ2.** What are the individual contributions of KG grounding and symbolic validation? **RQ3.** How effectively does the framework handle neuro-symbolic queries (routing, verification, latency)? **RQ4.** Does Annotated Datalog inference match a reference implementation and propagate probabilistic bounds correctly?

The **main benchmark** consists of 57 manually curated questions covering the institutional domain of a Brazilian public university across eight thematic categories (syllabi, faculty assignments, prerequisites, dependents, curriculum structure, coordinator information, contact information, and regulations), each annotated with a reference answer from official documents. Queries are routed to four specialized agents (*Disciplines*, *Faculty*, *Programs*, *Regulations*) and include both simple fact-retrieval questions and compositional multi-hop reasoning. A second **neuro-symbolic feature set** of 25 queries targets the five FOL rules: trajectory_planning (R5), critical_disciplines (R3), co_prerequisite (R4), prerequisite_chain (R1), and discipline_docentes (relational), with five queries each. The KG itself is built from 236 institutional markdown files (222 disciplines, 6 regulation files, 7 curriculum matrices, 1 faculty file), comprising 692 nodes and 1584 edges across 9 entity types (322 disciplines, 129 faculty, 108 areas, 50 FAQ, 42 regulation articles, 21 organizations, 7 programs, 7 curricula, 6 institutional documents).

4.1 Baseline and Protocol Configuration

We compare the framework against **four progressively stronger baselines**, all using the same model `ministral-3:8b` through Ollama and the same document corpus: (*B1*) LLM-only, answering from parametric knowledge with no retrieval; (*B2*) Standard RAG, hybrid retrieval (dense + BM25 + RRF) feeding contextual evidence to the LLM; (*B3*) Graph-RAG, hybrid retrieval augmented with KG-derived context, without symbolic validation; and (*B4*) Bidirectional Neuro-Symbolic Graph-RAG (ours), as previously presented in Section 3.

Responses on the main benchmark were independently evaluated by two annotators using three labels (*Correct*, *Partially Correct*, *Incorrect*); we report *Strict Accuracy* (Correct only) and *Weighted Accuracy* (wAcc = Acc+0.5 P, where P is the fraction labeled Partially Correct). Disagreements between the two annotators were adjudicated to consensus, so all reported counts are integers over 57; pre-adjudication inter-annotator agreement is $\kappa \approx 0.54$ (moderate) [Cohen 1960]. For the neuro-symbolic feature set, we additionally report *routing accuracy* (correct dispatch to the symbolic agent) and *symbolic verification* (structural correctness w.r.t. the KG). Latency is measured per query. Experiments run on a MacBook Air M1 (16 GB) with Ollama (`ministral-3:8b` for generation, `mxbai-embed-large` for embeddings), both served under Ollama's default quantization (Q4_K_M), which affects absolute accuracy and latency figures. PyReason runs under Numba JIT with a persistent cache.

4.2 Main Results (RQ1, RQ2)

Table II reports performance on the 57-question benchmark. The proposed framework (B4) achieves 84.2% strict accuracy, 87.7% weighted accuracy, and reduces the proportion of incorrect responses to 8.8%, a gain of 77.2 pp in strict accuracy and an 87% reduction in error rate relative to LLM-only. Standard retrieval alone (B2) brings only marginal improvements, indicating that many institutional queries require relational reasoning beyond document matching. Introducing KG grounding (B3) yields the largest single gain (strict accuracy 7.0%→57.9%), and the bidirectional neuro-symbolic layer (B4) produces a further substantial improvement, demonstrating that symbolic reasoning and validation complement graph grounding.

Ablation reading. Under `ministral-3:8b`, standard RAG has limited impact (B1→B2: +8.8 pp wAcc, no strict-accuracy gain), since prerequisite chains, faculty assignments, and curriculum structure are relational and resist text-similarity retrieval. KG grounding is the dominant factor (B2→B3: +32.4 pp wAcc, strict accuracy 7.0%→57.9%). Symbolic validation then provides the final reliability layer (B3→B4: +27.2 pp wAcc, error 36.8%→8.8%): corrective rewriting and the direct shortcut convert partially correct responses into fully correct ones, and the loop is not redundant with KG grounding, which B3 already exploits as context.

Table II. Benchmark performance on the 57-question benchmark: human-annotated accuracy comparing `ministral-3:8b` and `gemma4:12b`. Acc: strict accuracy; wAcc: weighted accuracy; Err: incorrect answer rate. Ministral columns: two annotators; Gemma4 columns: single annotator.

	Ministral			Gemma4		
System	Acc	wAcc	Err	Acc	wAcc	Err
B1 - LLM-only	7.0%	19.3%	68.4%	0.0%	5.3%	89.5%
B2 - Standard RAG	7.0%	28.1%	50.9%	14.0%	50.9%	12.3%
B3 - Graph-RAG	57.9%	60.5%	36.8%	40.4%	68.4%	3.5%
B4 - NS Graph-RAG (ours)	**84.2%**	**87.7%**	8.8%	77.2%	**88.6%**	**0.0%**

Table III. Routing accuracy and symbolic verification by neuro-symbolic feature (FOL rule).

Feature	**Rule**	**Routed correctly**	**Verified**
trajectory_planning	R5	5/5 (100%)	5/5 (100%)
critical_disciplines	R3	5/5 (100%)	4/5 (80%)
co_prerequisite	R4	5/5 (100%)	5/5 (100%)
prerequisite_chain	R1	5/5 (100%)	5/5 (100%)
discipline_docentes	—	5/5 (100%)	5/5 (100%)
Total	—	**25/25 (100%)**	**24/25 (96%)**

4.3 Neuro-Symbolic Feature Evaluation (RQ3)

Table III reports routing and verification on the 25-query neuro-symbolic feature set. Routing reaches 100%, and symbolic verification reaches 96%, with the single failure on *critical_disciplines* corresponding to a borderline node whose classification depends on the threshold $\theta = 3$ rather than a reasoning failure. R1, R4, and R5 are perfectly verified. **Latency.** On 82 queries spanning all paths (`ministral-3:8b` backend), the symbolic path averages 1.56 s against 4.76 s for neural paths, a 3.1× speedup. The gain is structural: the direct shortcut converts an LLM call into a KG traversal independently of model size.

4.4 PyReason Engine: Parity and Bounds Propagation (RQ4)

To validate the symbolic layer independently of the end-to-end pipeline, we compared the PyReason implementation of R1 and R4 against a pure-Python reference on 8 transitive-closure samples and 5 co-prerequisite samples from the KG: *the two implementations agree on every output* (zero divergence). On the synthetic chain $A \to_{1.0} B \to_{0.7} C \to_{1.0} D$, PyReason returns prereq_trans$(A, D) = [0.7, 1.0]$ as expected; introducing a parallel curated chain restores the bound to $[1.0, 1.0]$, confirming the max-bottleneck-path semantics. These tests confirm the formal claims of the symbolic layer and rule out a class of implementation defects.

To complement human annotation, we conducted an independent LLM-as-judge evaluation on 38 semantic queries (with reference answers as textual descriptions), using `deepseek-v3.2` as the judge, a different model family from the responder, with greedy decoding (temperature 0) and the full judge prompt shipped with the released code, scoring (1-5 scale) *Groundedness*, *Correctness*, *Completeness*, and *Clarity*. The judge received both the system response and the retrieved evidence and was instructed to score only against that evidence. With the `ministral-3:8b` backend, the framework reaches high overall averages: 4.79 groundedness, 4.79 correctness, 4.74 completeness, and 5.00 clarity (Table IV). Regulations and symbolic-KG paths achieved ideal scores; the Disciplines path is weakest in completeness (4.18), reflecting longer-form descriptive answers for syllabus queries. Re-running the same judge protocol with the `gemma4:12b` yields overall scores of 4.94 / 4.90 / 5.00 / 5.00, slightly higher than the ministral and confirming that semantic-path quality is preserved under a different, larger backend. Results converge with the human evaluation in Table II: the weakest dimension (completeness) corresponds to cases where the system correctly abstains rather than fabricating.

Table IV. LLM-as-judge scores by reasoning path under the `ministral-3:8b` backend (38 semantic queries, scale 1-5).

Path	*N*	**Ground.**	**Correct.**	**Complete.**	**Clarity**
Disciplines	11	4.45	4.45	4.18	5.00
Faculty	11	4.82	4.82	4.91	5.00
Regulations	5	5.00	5.00	5.00	5.00
Symbolic KG (semantic)	11	5.00	5.00	5.00	5.00
Overall	**38**	**4.79**	**4.79**	**4.74**	**5.00**

Table V. Qualitative responses to the compositional query "*What do I need to take before Compilers?*" (Gemma4 backend, single-annotator labels). Responses abridged for space.

Sys	**Label**	**Response (abridged)**
B1	I	"To take Compilers, you should master: 1. Data Structures; 2. Formal Languages; 3. Programming; ..." *(generic topics, no institutional prerequisite)*
B2	P	"To take Compilers, you must have completed: Formal Languages and Automata (Code: 2616)." *(direct prerequisite only)*
B3	P	"To take Compilers, you need to have completed: Formal Languages and Automata; Discrete Mathematics; Logic Programming." *(flat list, no chain)*
B4	C	"**Prerequisite chain of Compilers:** Compilers ← Formal Languages and Automata ← Discrete Mathematics ← Logic Programming. To take **Compilers**, you must have completed: Formal Languages and Automata; Discrete Mathematics; Logic Programming."

4.5 Backend Sensitivity

To verify that our results do not depend on a specific LLM backend, we ran the full pipeline replacing `ministral-3:8b` with `gemma4:12b`, a larger model run locally under the same Ollama setup and default quantization, and re-annotated the 57-question benchmark under the same C/P/I protocol. Table II reports human-annotated accuracy for both backends side by side. The ordering B1<B2<B3<B4 in weighted accuracy is preserved, and B4 again outperforms all baselines, reaching 88.6% wAcc with no answer labeled Incorrect (0/57 under the single-annotator protocol), against 3.5% for the Graph-RAG baseline and 89.5% for LLM-only. The B3→B4 wAcc gain remains substantial (+20.2 pp), confirming that symbolic validation contributes beyond what graph grounding alone provides under a different, larger backend. We further note that B1 (LLM-only) shows no improvement under the larger backend (Err 89.5% vs. 68.4%, across different annotation protocols): scaling the model without retrieval does not close the institutional-knowledge gap. Absolute values are not strictly comparable across the two columns, but the relative trends are preserved, though the weight of each stage shifts: under `gemma4:12b` the B1→B2 step accounts for a much larger share of the total gain (+45.6 pp wAcc) than under `ministral-3:8b`.

As a qualitative example, Table V illustrates the responses of B1-B4 to the compositional query "*What do I need to take before Compilers?*" (ground-truth chain: *Compilers ← Formal Languages and Automata ← Discrete Mathematics ← Logic Programming*). B1 (LLM-only) hallucinates irrelevant topics; B2 (standard RAG) yields only the direct prerequisite; B3 (Graph-RAG) retrieves all three as a flat list; whereas B4 (neuro-symbolic) leverages a symbolic shortcut to output the explicit dependency chain with a KG-verification footer.

5. FINAL REMARKS

We presented a bidirectional neuro-symbolic validation framework for Graph-RAG that integrates KG grounding, first-order logic reasoning, Annotated Datalog inference with confidence bounds, and a bidirectional control loop combining symbolic enrichment, post-generation validation with corrective rewriting, and direct symbolic execution. On an academic-advisory benchmark from a Brazilian public university, the framework achieves 84.2% strict accuracy, 87.7% weighted accuracy, an 87% relative

reduction in factual errors compared to LLM-only, and a 27.2 pp gain over Graph-RAG without symbolic validation. Routing on neuro-symbolic queries reaches 100%, symbolic verification 96%, the symbolic path is 3.1× faster than neural paths, and PyReason matches a pure-Python reference with zero divergence while propagating mixed-confidence bounds correctly.

The results show that combining a small set of FOL rules, a logic engine, and bidirectional validation substantially closes the neural retrieval reliability gap in graph-structured domains. A backend-sensitivity study using the larger `gemma4:12b` left no B4 answer labeled Incorrect (0/57) while yielding no improvement for the LLM-only baseline (B1), suggesting that model scaling alone cannot bridge the institutional-knowledge gap.

Limitations: the benchmark covers 57+25 queries from one institution, and B4's zero observed errors under `gemma4:12b` (single annotator) are compatible with true rates up to ∼5% at 95% confidence for n=57. The five FOL rules are domain-generic dependency patterns (transitivity, gating, centrality, co-dependency, planning), so porting to other DAG domains means re-instantiating predicates and thresholds rather than new reasoning machinery, at a manual cost still to be quantified. The backend comparison varies size, family, and quantization together, a robustness check rather than a controlled scaling ablation. Future work will explore temporal/preference constraints, other DAG domains, and the effect of symbolic explanations on user trust.

Code Availability: The source code and the complete implementation are publicly available at `https://github.com/Arazoleo/FESP-AI`.

REFERENCES

Aditya, D., Mukherji, K., Balasubramanian, S., Chaudhary, A., and Shakarian, P. Pyreason: Software for open world temporal logic. In *AAAI Spring Symposium on Challenges Requiring the Combination of Machine Learning and Knowledge Engineering*, 2023.

Brown, A., Roman, M., and Devereux, B. A systematic literature review of retrieval-augmented generation: Techniques, metrics, and challenges. *arXiv preprint arXiv:2508.06401*, 2025.

Cheng, M., Luo, Y., Ouyang, J., Liu, Q., Liu, H., Li, L., Yu, S., Zhang, B., Cao, J., Ma, J., Wang, D., and Chen, E. A survey on knowledge-oriented retrieval-augmented generation. *arXiv preprint arXiv:2503.10677*, 2025.

Cohen, J. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* 20 (1): 37–46, 1960.

Garcez, A. S. d. and Lamb, L. C. Neurosymbolic ai: The third wave. *Artificial Intelligence Review* vol. 56, pp. 12387–12406, 2023.

Han, H., Ma, L., Wang, Y., Shomer, H., Lei, Y., Qi, Z., Guo, K., Hua, Z., Long, B., Liu, H., Aggarwal, C. C., and Tang, J. Rag vs. graphrag: A systematic evaluation and key insights. *arXiv preprint arXiv:2502.11371*, 2025.

Linders, J. and Tomczak, J. M. Knowledge graph-extended retrieval augmented generation for question answering. *Applied Intelligence* vol. 55, pp. 1102, 2025.

Saha, A., Agarwal, P., et al. A zero-shot neuro-symbolic approach for complex knowledge graph question answering. *arXiv preprint arXiv:2601.04568*, 2026.

Saxena, Y. and Gaur, M. Neurosymbolic retrievers for retrieval-augmented generation. *IEEE Intelligent Systems*, 2026.

Wu, J., Zhu, J., Qi, Y., Chen, J., Xu, M., Menolascina, F., and Grau, V. Medical graph rag: Towards safe medical large language model via graph retrieval-augmented generation. *arXiv preprint arXiv:2408.04187*, 2024.

Yu, D., Yang, B., Liu, D., Wang, H., and Pan, S. A survey on neural-symbolic learning systems. *Neural Networks* vol. 166, pp. 105–126, 2023.

Zhang, Q., Chen, S., Bei, Y., Yuan, Z., Zhou, H., Hong, Z., Chen, H., Xiao, Y., Zhou, C., Dong, J., Chang, Y., and Huang, X. A survey of graph retrieval-augmented generation for customized large language models. *arXiv preprint arXiv:2501.13958*, 2025.

Zhu, X., Xie, Y., Liu, Y., Li, Y., and Hu, W. Knowledge graph-guided retrieval augmented generation. In *Proceedings of the 2025 Annual Conference of the Nations of the Americas Chapter of the ACL (NAACL 2025)*, 2025.