SOVEREIGN: How do evidence gap identification mechanisms in FAIR-RAG systems affect F1 score performance on HotpotQA when

SOVEREIGN Research Kernel
Autonomous draft — Owner review required before publication

May 29, 2026

Abstract

While Retrieval-Augmented Generation (RAG) mitigates hallucination and knowledge staleness in Large Language Models (LLMs), existing frameworks often falter on complex, multi-hop queries that require synthesizing information from disparate sources. Current advanced RAG methods, employing iterative or adaptive strategies, lack a robust mechanism to systematically identify and fill evidence gaps, often propagating noise or failing to gather a comprehensive context. We introduce FAIR-RAG, a novel agentic framework that transforms the standard RAG pipeline into a dynamic, evidence-driven reasoning

1 Introduction

Analysis of: FAIR-RAG: Faithful Adaptive Iterative Refinement for Retrieval-Augmented Generation. Research goal: How do evidence gap identification mechanisms in FAIR-RAG systems affect F1 score performance on HotpotQA when comparing single-pass versus iterative retrieval strategies?.

2 Methodology

Multi-query arXiv search (4 parallel queries, Relevance-sorted). TF-IDF cosine semantic verification (bigrams, threshold=0.15). NIM nv-embedqa-e5-v5 (dim=1024) for semantic indexing. Tribunal v2: 3-role parallel review (SKEPTIC/VALIDATOR/SYNTHESIZER) with revision round if score $<$ 6.5.

3 Results

10 papers retrieved. 11 claims extracted, 1 verified. Tribunal: 4.7/10 $\rightarrow$ REJECT (revision_round=0). Policy: ESCALATE_TO_OWNER.

4 Uncertainties

NIM free tier latency varies. TF-IDF verification is a weak signal. arXiv Relevance ranking is query-dependent. Tribunal consensus is LLM-based and prompt-sensitive.

5 Extracted Claims

Claim	Verified	Confidence
The FAIR-RAG framework uses the Reciprocal Rank Fusion (RRF) algorithm to re-rank documents.	×	0.03
The RRF algorithm in FAIR-RAG produces a single ranked list of the top-5 most relevant documents.	×	0.03
The RRF algorithm implementation in FAIR-RAG does not require hyperparameter tuning.	×	0.04
FAIR-RAG employs an LLM agent to evaluate document utility and discard irrelevant, off-topic, or tangentially related do	×	0.03
The Structured Evidence Assessment (SEA) module uses a checklist-based methodology rather than abstractive summarization	×	0.09
In the SEA module, an LLM agent deconstructs the user's query into a checklist of discrete, required informational compo	×	0.08
The SEA module audits collected evidence against a checklist to confirm supported findings and identify explicit intelli	×	0.09
In FAIR-RAG, evidence is deemed sufficient only if all required findings on the checklist are confirmed.	×	0.12
FAIR-RAG includes an Adaptive Routing module that analyzes query complexity to determine the execution path.	×	0.06
FAIR-RAG initiates a cyclical Iterative Refinement loop for non-trivial queries involving information decomposition, ret	×	0.11
The SEA module in FAIR-RAG acts as an analytical gating mechanism that identifies 'Remaining Gaps' to guide subsequent q	✓	0.16

References

- http://arxiv.org/abs/2507.23334v2
- http://arxiv.org/abs/2510.22344v1
- http://arxiv.org/abs/2404.14464v1