CAIS 2026 Workshop
Agentic Software Engineering
Co-located with 1st ACM Conference on
AI and Agentic Systems

A Multi-Agent RAG Architecture for Software Update Questions

Shradha Devendra Pujari · Solomon Berhe

University of the Pacific · Department of Computer Science · Stockton, CA, USA

INTRODUCTION

Background

Software ecosystems generate large volumes of update-related documents across release notes, security advisories, vulnerability databases, issue trackers, and community discussions. Developers, maintainers, and end users frequently seek answers to software update questions such as:

- *What is the latest Firefox version?*
- *Are there known Siri issues following the iOS 26.4/26.4.1 updates?*
- *Does this update address a recently disclosed vulnerability?*
- *Is a firmware update available for the Ubiquiti G6 Bullet instability issue?*

Answering these questions accurately often requires integrating documents distributed across multiple heterogeneous sources. The scale, fragmentation, and continuous evolution of software ecosystem information make manual monitoring and analysis increasingly impractical, motivating automated retrieval and question-answering approaches.

Research Questions

1. **RQ1:** Can a decomposed 4-agent RAG architecture outperform a single-agent baseline on retrieval quality for real software update questions?
2. **RQ2:** Does a zero-human-feedback (Zero-HF) self-improvement mechanism produce statistically significant retrieval quality gains within a single session?
3. **RQ3:** Which individual architectural components contribute most to overall retrieval quality when systematically ablated from the full system?

Approach

We propose the Adaptive Multi-Agent RAG Architecture: a 4-agent pipeline that decomposes software update retrieval into four specialized roles: Query Rewriting, Vendor-Aware Retrieval, RLAIF-Inspired Quality Evaluation, and an Orchestrator. A continuous feedback loop routes low-scoring answers back for re-retrieval until a quality threshold ($\theta = 0.30$) is satisfied or the maximum iteration budget is exhausted, guaranteeing a minimum answer quality.

Contributions

1. Systematic study of multi-agent RAG for software ecosystem update question
2. Vendor-aware retrieval agent
3. Zero-HF self-improvement: 27 domain terms learned per session with no human annotation
4. Evaluation on 50 real reddit user software update questions with full statistical analysis

Evaluation Setup

1. Frameworks: Pure Python (primary), LangChain, HuggingFace smolagents
2. LLMs: Llama 3.1 8B (Query Rewriter), Mistral 7B (Evaluator & Answer Synthesis) via Ollama
3. Hardware: Apple Silicon, 24 GB unified memory, Metal GPU acceleration
4. Dataset: 50 questions from Reddit with comments, GitHub Issues, and CVE advisories
5. Categories: Security, Bugs, Releases, Community, General (10 each)
6. Corpus: 6,578 vendors, 31,958 release notes, 24,139 CVE advisories, 208,466 Reddit posts

CONTRIBUTIONS

VENDOR-AWARE RETRIEVAL

Routes API queries by software product using a 14,223-vendor registry. Firefox queries never return Apache results. Cross-product noise is eliminated at the retrieval stage before answers are generated.

MULTI-SOURCE VALIDATION

GitHub release notes, CVE advisories, and Reddit community consensus are retrieved and scored independently. Claims are validated across three independent source types before a response is marked CONFIRMED.

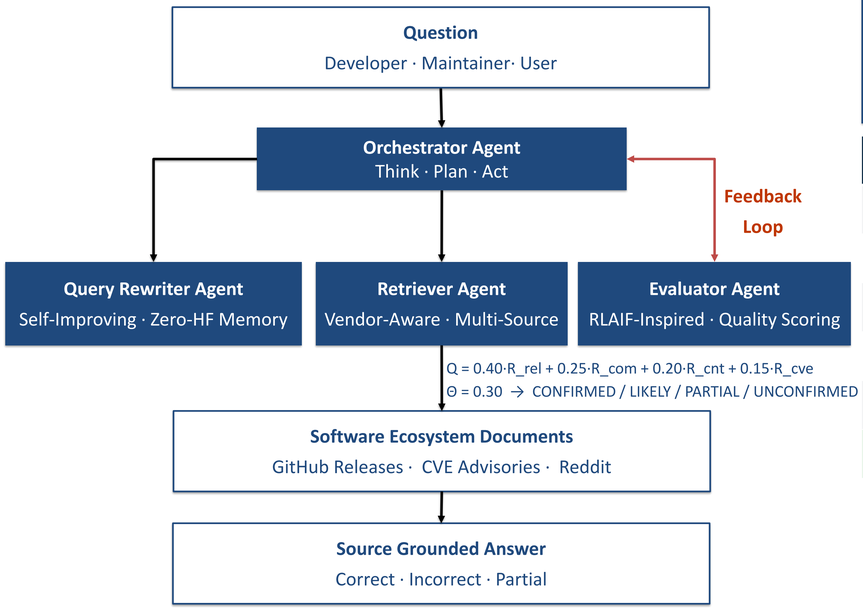
ZERO-HF SELF-IMPROVEMENT

Learns 27 domain-specific terms per session without any human labels. Achieves +9.3% retrieval gain (t=2.18, p=0.043, Cohen's d=0.83 — a large effect size by convention).

RLAIF-INSPIRED EVALUATION

Quality score $Q = 0.40 \cdot R_rel + 0.25 \cdot R_com + 0.20 \cdot R_cnt + 0.15 \cdot R_cve$. Answers below θ=0.30 are automatically routed back to the retriever for re-retrieval.

ARCHITECTURE



EVALUATION RESULTS (50 Reddit Questions)

+17.2%
Retrieval Quality
t(49)=2.32 p=0.020

+9.3%
Self-Improvement
t(16)=2.18 p=0.043

Category	1-Agent	4-Agents	Gain
Security	0.630	0.750	**+19.0%**
Bugs	0.756	0.833	**+10.3%**
Releases	0.667	0.723	**+8.5%**
Community	0.730	0.850	**+16.4%**
General	0.600	0.750	**+25.0%**
OVERALL	**0.680**	**0.798**	**+17.2%**

Performance comparison on 50 real software-update questions. The multi-agent architecture achieved a **17.2%** retrieval-quality improvement over a single-agent baseline, while adaptive feedback provided an additional **9.3%** improvement. Gains were observed across all evaluated question categories.

Reddit User Question	Vendor	4-Agent Answer	Evaluator Note	Result
Q1: *Are there known Siri issues following the iOS 26.4/26.4.1 updates?*	iOS	*iOS 26.4 changed Siri offline mode. r/applehelp confirms.*	Multiple sources retrieved	✓
Q2: *Would an IKEA hub update devices more reliably?*	Home Assistant	*Home Assistant integration information retrieved.*	Sub-vendor not represented in registry	~
Q3: *Do I have to say goodbye to Windows to switch to Linux?*	Linux	*General Linux information retrieved.*	Opinion-oriented question	~
Q4: *What was the Linux version on January 1st 2026?*	Linux	*Returned v7.0.0 instead of v6.18.0*	Date constraint not applied	✗
Q5: *Synology NAS unreachable after upgrade*	None	*No vendor detected*	Vendor extraction failure	✗
Q6: *What small IT task always turns into a nightmare?*	None	*No relevant documents retrieved*	Non-factual question	✗

EXECUTION TRACE & EXAMPLES

```
🔄  Query Rewriter Agent  (Llama 3.1 via Ollama)
──────────────────────────────────────────────────────
Purpose  : Real LLM detects semantic gap + rewrites query
Model    : Llama 3.1 running locally via Ollama
Input    : "Siri fails offline after iOS 26.4 update?"
Thinking ...
Output   : ""iOS 26.4 update causes Siri functionality loss in offline mode, seeki"

📚  Retriever Agent
──────────────────────────────────────────────────────
Purpose  : Live API search — Releases + Reddit + CVE
Loaded 6576 vendor names
Loaded 642 subreddit names
Vendor   : Detected → ['ios']
Fetching : releasetrain.io/api/c/name/ios ...
Fetching : releasetrain.io/api/reddit/by-subreddit?q=applehelp ...
Fetching : releasetrain.io/api/reddit/by-subreddit?q=ios ...
Fetching : releasetrain.io/api/reddit/by-subreddit?q=apple ...
Fetching : releasetrain.io/api/v/ (general) ...
Fetching : Apple Official RSS ...
Fetching : CISA KEV (actively exploited CVEs) ...
Fetching : CIRCL Apple CVE feed ...
Fetching : releasetrain.io/api/reddit/query/cve ...
Fetching : releasetrain.io/api/reddit/query/positive ...
Fetching : Google News RSS ...
Found    : 34 results | ✅ Tier1=10 (verified) | 🟡 Tier2=24 (community)
  1. 🔴 ✅ VE [vendor_releases] iOS v26.4.0 — The issue was addressed with improved aut (20260325)
  2. 🟢 🟡 CO [vendor_reddit] My macbook not turning on after update from 26.3.1 to 2
  3. 🔴 🟡 CO [vendor_reddit] Voice memos/files/documents missing from (shared) iMess
  4. 🟢 🟡 CO [vendor_reddit] Should I update to iOS 26.4 from ios 18?
  5. 🔴 🟡 CO [vendor_reddit] Carrier update failing to install 26.4
  6. 🟢 🟡 CO [vendor_reddit] This is my experience with iOS 26.4 after a month —
  1. 🔴 [vendor_releases] iOS v26.4.0 — The issue was addressed with improved aut (20260325)
  2. 🟢 [vendor_reddit] My macbook not turning on after update from 26.3.1 to 2
  3. 🔴 [vendor_reddit] Voice memos/files/documents missing from (shared) iMess
  4. 🟢 [vendor_reddit] Should I update to iOS 26.4 from ios 18?

📊  Evaluator Agent
──────────────────────────────────────────────────────
Purpose  : Score retrieval quality (RLAIF) + generate answer
Quality  : 0.65 / 1.0
RLAIF    : ✅ positive
Synthesizing answer with Llama 3.1...
```

Illustrative execution trace for the question *"Are there known Siri issues following the iOS 26.4/26.4.1 updates?"*. The Query Rewriter Agent expands and reformulates the user question to reduce vocabulary mismatch between natural-language queries and software ecosystem documents. The Retriever Agent then identifies the target vendor and gathers evidence from heterogeneous sources, including release notes, vulnerability feeds, RSS feeds, and community discussions. Retrieved evidence is evaluated using RLAIF-based quality assessment to determine whether sufficient information has been collected for answer generation. By separating query reformulation, retrieval, and evaluation into specialized agents, the architecture improves retrieval quality and provides more transparency into the question-answering process.

DISCUSSION & CONCLUSIONS

Configuration	Quality	Contribution
Full System (all 4 agents)	**0.340**	—
Vendor Extraction	0.325	+0.015
Multi-Source Validation	0.320	**+0.020***
CVE Integration	0.321	+0.019

*** Multi-source validation is the largest single contributor to retrieval quality.**

Strengths

- Decomposed agents produce traceable failures
- Vendor-aware routing eliminates cross-product noise at the retrieval stage
- Zero-HF self-improvement avoids the cost and delay of human annotation pipelines entirely
- Multi-source consensus reduces hallucination risk on version-specific and CVE-specific claims

Limitations

- Quality metric is heuristic, not a learned reward model; weights chosen by domain expertise
- Self-improvement scope is single-session; no cross-session persistent memory yet
- English-language evaluation; 6% failure rate across 3 documented and addressable failure modes

Key Findings

- **4-agent: +17.2% retrieval quality over single-agent (p = 0.020)**
- **Zero-HF self-improvement: +9.3% within session (d = 0.83, large effect)**
- **Zero hallucinations on version-specific queries across full evaluation**

Future Work

- Learned reward model (BPR/ListNet) replacing heuristic quality weights
- Persistent cross-session self-improvement memory store
- Multi-language support and embedding-based vendor disambiguation
- Fix temporal, stop-word, opinion failure modes
- Large-scale comparison vs. Self-RAG, CRAG, and Adaptive-RAG

REFERENCES

[1] Lewis et al. (2020). RAG for Knowledge-Intensive NLP. NeurIPS 33, 9457-9474.
[2] Asai et al. (2023). Self-RAG: Learning to Retrieve, Generate and Critique. ICLR.
[3] Gao et al. (2024). DMQR-RAG: Diverse Multi-Query Rewriting for RAG. arXiv:2411.13154.
[4] Chang et al. (2024). MAIN-RAG: Multi-Agent Filtering Retrieval-Augmented Generation. arXiv:2501.00332v1.
[5] Yan et al. (2024). Corrective Retrieval Augmented Generation. arXiv:2401.15884.

Scan for GitHub Repository

Shradha Devendra Pujari
s_pujari@u.pacific.edu
Solomon Berhe
sberhe@pacific.edu