Tavily Web Summarization

Production-Scale 2-Agent Architecture

Ofir Suranyi | `github.com/ophirshurany/Tavily` | January 2026

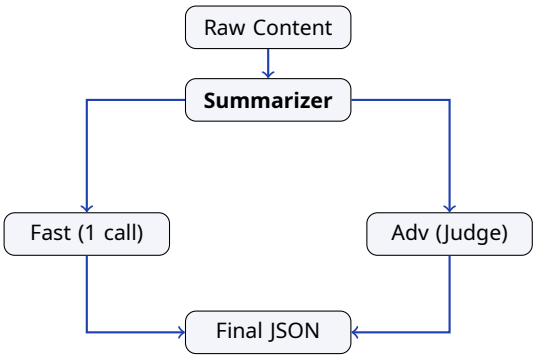
1. Executive Overview

This project optimizes web content summarization for high-throughput production (millions of daily requests). We transitioned from a sequential 3-agent baseline to a consolidated **2-Agent Model** leveraging **Gemini 2.0 Flash**, achieving a **50% latency reduction** and **33–50% cost savings** without degrading semantic quality.

2. Technical Architecture

Consolidated Workflow:

- **Summarizer Agent:** Merges Research + Writing into one LLM call using Chain-of-Thought (CoT).
- **Judge Agent (Advanced Path):** Validates output using Pydantic schema enforcement and a weighted score: $Q = 0.4E_c + 0.3C_c + 0.3C_h$.
- **Reliability:** Asyncio Semaphores manage rate limits (500 RPM); exponential backoff handles transient failures.



3. Production Benchmarking (N=950 Samples)

Strategy	Latency	BERTScore	Cost/Req	Throughput	LLM Calls
3-Agent (Base)	15–20s	0.88	$0.0040	1x	3
Fast Strategy	**2–4s**	0.82	**$0.0008**	**5x**	**1**
Advanced Strategy	8–12s	**0.90**	$0.0020	3x	2

4. Key Engineering Implementations

- **Pydantic Schema Enforcement:** Guarantees structured JSON output, critical for downstream API integration.
- **Multilingual Support:** Native handling for 50+ languages; Judge enforces "Source-to-Target" language consistency.
- **Concurrency Engine:** Python `asyncio` implementation allows 50–100 concurrent requests, maximizing API usage.

5. Constraints & Mitigation

- **Hallucinations:** ROUGE-L fails to detect factual errors; QAGS (Phase 2) will verify facts via automated Q&A.
- **Context Limits:** Hard truncation at 8k chars for Gemini Flash. Hierarchical summarization planned for docs >20k tokens.

6. Next Steps for Production Growth

1. **Redis Caching:** Aiming for 40% cost reduction by caching trending content.
2. **Distillation:** Fine-tune a local **Llama-3.2-8B** on Advanced logs for sub-1s, zero-cost summarization.

Takeaway: The 2-agent architecture consolidates "thinking" steps into a single context, providing the optimal speed-to-accuracy ratio for real-time web indexing.