

IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

VOLUME : 11 ISSUE : 07 Print / Issue Publication Date: November 2026

ISSN : 2455-2143

DOI : 10.33564/IJEAST.2026.v11i07.005

Indexed In

Google SCHOLAR OPEN ACCESS doi Crossref

WWW.IJEAST.COM

editor@ijeast.com

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 07, ISSN No. 2455-2143, Pages 32-42
Published Online November 2026 in IJEAST (http://www.ijeast.com)

TESSERACT: AN AUTONOMOUS 4-DIMENSIONAL MULTI-AGENT INTELLIGENCE ENGINE FOR VERIFIABLE DEEP RESEARCH AND ZERO HALLUCINATION SYNTHESIS

Afroz Khan, Shrikrishna Bhosale, Shruti Bokade
Department of Computer Science and Engineering,
MIT College of Railway Engineering & Research,
Barshi, Maharashtra, India

Prof. Somnath Mule
Assistant Professor, Department of Computer Science and Engineering,
MIT College of Railway Engineering & Research,
Barshi, Maharashtra, India

Abstract— **Compiling technical research from uncurated web sources requires evaluating dozens of conflicting articles and extracting empirical data points. Standard Large Language Models (LLMs) and basic Retrieval Augmented Generation (RAG) tools invent statistics in long texts, lose technical context through naive vector chunking, and produce citations that point to broken links or do not support the text. To resolve these problems, we built Tesseract (v2.3.0), an autonomous multi-agent research system that coordinates five specialized agents over a state graph using LangGraph. Tesseract breaks inquiries into four dimensions: temporal horizon, structural depth, domain specialization, and adversarial verification. A Query Normalizer cleans the input, a Strategic Planner creates bounded sub-queries, a Web Scout gathers and canonicalizes web pages while scoring domain authority, an Adversarial Auditor checks evidence coverage and verifies claims using Natural Language Inference (NLI), and a Synthesis Editor produces the final report. Groq Language Processing Units (Llama 3.3 70B, over 250 tokens per second) handle planning and auditing, while Google Gemini 2.5 Flash handles long-form writing, completing 3,500-word reports in under 100 seconds. Across more than 50 technical benchmark topics, the system achieved a 98.4% claim grounding rate and zero ghost citations. The entire stack runs on free-tier infrastructure.**

Keywords— **Multi-agent systems, LangGraph, deep research, large language models, factual grounding, hallucination elimination, Groq LPU, information retrieval, automated technical synthesis.**

I. INTRODUCTION

Engineers, clinical investigators, and academic researchers spend 30% to 40% of their working hours aggregating and reconciling technical documentation dispersed across digital archives [1]. Search engines return thousands of rank-ordered landing pages, while conversational AI gives succinct answers that obscure trade-offs, omit primary sources, and smooth over technical nuance. Building an exhaustive, multiperspective dossier on a cutting-edge topic remains a manual process of reading, cross-referencing, and bibliographic validation.

Using generative LLMs directly to automate this synthesis exposes four structural failure modes that stem from their next-token probabilistic architecture.

Unconstrained hallucination. LLMs generate prose by estimating conditional probability distributions over tokens [2], [17]. In multi-thousand-word texts, attention diffuses across earlier outputs and the model shifts from factual recall to plausible narrative completion, inventing benchmark metrics, fabricating clinical trial phases, and stating conjecture as verified fact [2], [14].

Decontextualization from naive vector chunking. Standard RAG splits crawled pages into fixed windows of 250 to 500 tokens and retrieves the top-k chunks by cosine similarity [3], [6], [22]. This shreds comparison tables into row fragments, divorces qualifications from governing equations, and yields a repetitive, superficial listicle.

Phantom citations and dead links. Between 30% and 50% of citations generated by commercial chatbots point to HTTP 404 pages, resolve to ad redirects, or fail to substantiate the sentence they accompany [4], [23]. Models

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 07, ISSN No. 2455-2143, Pages 32-42
Published Online November 2026 in IJEAST (http://www.ijeast.com)



also exhibit "citation borrowing," pairing a valid URL with a fabricated number.

Monolithic attention drift. In an unconstrained ReAct loop [7], raw HTML, error pages, and intermediate thoughts accumulate in one context buffer. The model forgets its plan, fixates on keyword tangents, and repeats searches that burn API quota without finding new evidence [7], [21].

We therefore developed Tesseract (v2.3.0), which treats research as a deterministic, state-governed pipeline rather than a conversational chat. Orchestrated with LangGraph [5], it routes each inquiry through five agent nodes for intake normalization, search decomposition, source authority scoring, adversarial verification, and editorial publication. The main contributions are:

(1) **Decoupled 5-agent state graph:** planning, web harvesting, fact checking, and synthesis run in isolated nodes with strict input/output contracts, which removes attention drift.

(2) **4-dimensional research state model:** temporal horizon, structural granularity, domain specialization (8 archetype outlines), and adversarial verification.

(3) **Authority-tiered scraping engine:** strips tracking parameters (utm_*, fbclid, gclid), unwraps redirects, removes fragment anchors, filters personal anecdotes, and ranks sources in three tiers.

(4) **Adversarial Auditor with closed-loop reflection:** an evidence-coverage gate ($C_{cov} \geq 0.85$) and NLI claim verification ($\tau = 0.82$), with at most $C_{max} = 2$ reflection cycles.

(5) **Deterministic anti-slop post-processor:** repairs single row tables, removes robotic clichés, balances code fences, and enforces a zero-ghost-citation invariant.

(6) **Dual-engine acceleration and benchmarking:** Groq LPU (Llama 3.3 70B) with Gemini 2.5 Flash on free-tier infrastructure, evaluated on 50 topics against GPT-4o, naive RAG, and Perplexity Pro.

II. RELATED WORK

Retrieval-augmented generation. Early neural question answering relied on parametric weights alone [17]–[19], which suffer from knowledge cutoffs and hallucination on niche topics. Lewis et al. [3] introduced RAG, pairing a sequence-to-sequence model with a dense index queried by Maximum Inner Product Search. Karpukhin et al. [6] added Dense Passage Retrieval, and Guu et al. [22] integrated retrieval into pre-training (REALM). Yet standard RAG remains brittle for long-form research: fixed windows sever structure, and retrieval is single-shot, with no check that the passages answer the question [3].

Agentic reasoning and reflection. ReAct [7] interleaves reasoning traces with tool calls, Toolformer [20] trains models to choose APIs, and Reflexion [21] adds verbal self reflection. In a single monolithic context, however, the scratchpad saturates with snippets and HTML, causing reasoning degradation and context drift [7], [21].

Multi-agent frameworks. AutoGen [8] uses conversational multi-agent patterns, and MetaGPT [9] encodes standard operating procedures as software artifacts. Talebirad and Nadiri [24] show that deterministic state machines outperform unconstrained chatter by enforcing finite execution bounds and predictable schemas.

Automated deep research. STORM [10] simulates multi perspective interviews to write Wikipedia-style articles, and WebGPT [23] fine-tunes models with human feedback to browse and cite. Bubeck et al. [25] document emergent capabilities of foundation models. These systems rely on simulated personas or post-hoc self-critique that do not systematically verify claims against primary evidence, so an early error propagates. Tesseract replaces role-play with an NLI-backed closed-loop gate and strict URL canonicalization.

Inference hardware and long context. Open weights such as Llama 3 [11] democratized custom inference. Groq's Language Processing Unit [12] keeps weights in on-chip SRAM (over 80 TB/s) with deterministic scheduling, sustaining over 250 tokens/s on 70B models. Gemini 2.5 Flash [13] offers a context window above one million tokens, allowing single-pass synthesis over hundreds of sources without destructive chunking.

TABLE I. LITERATURE REVIEW AND TAXONOMY OF RETRIEVAL-AUGMENTED AND MULTI AGENT SYSTEMS

Framework & Ref.	Core mechanism	Identified limitation
RAG [3]	MIPS dense search, BART generator	Chunking shreds data matrices
ReAct [7]	Interleaved reasoning and tool actions	Monolithic scratchpad saturates
AutoGen [8]	Multi-agent conversational role play	Unbounded chat loops inflate cost

MetaGPT [9]	SOPs as structured software artifacts	Built for dev workflows, not research
STORM [10]	Pre-writing perspective interviews	Persona errors propagate unchecked
Groq LPU [12]	Deterministic TSP, SRAM >80 TB/s	Raw hardware; needs orchestration
Gemini 2.5 [13]	Native 1M+ token context	Multi-hop loops add latency
Framework & Ref.	**Core mechanism**	**Identified limitation**
TRUE NLI [14]	Cross-encoder entailment checks	Costly without heuristics
Trafilatura [15]	Heuristic main-text scraper	No source authority tiering
Tesseract (proposed)	5-agent LangGraph DAG, closed-loop NLI gate	Needs dual-engine API keys

Comparative analysis. Vanilla GPT-4o [25] relies on parametric weights and reaches only a 61.4% claim grounding rate with a 22.8% phantom citation rate. Naive RAG [3] fractures tables across token boundaries, producing garbled or incomplete rows in 12.0% of generated tables, and treats blog commentary like peer-reviewed work. Commercial agents such as Perplexity Pro and OpenAI Deep Research search iteratively, but run on centralized GPU clusters with latencies of 10 to 30 minutes per deep query and subscription fees up to $200.00/month. Tesseract reaches comparable factual density (3,640 words, 15.6 sources) and higher grounding (98.4% vs 91.2%) in under 50 seconds at Rs. 0.00 marginal infrastructure cost (Table II).

TABLE II. DEEP RESEARCH ARCHITECTURAL COMPARISON: TESSERACT VS COMMERCIAL STATE-OF-THE-ART

Dimension	**Baselines (GPT-4o / RAG / OpenAI)**	**Tesseract (proposed)**
Orchestration	Feedforward / single-shot / tree search	5-agent deterministic DAG
Inference	Standard cloud GPU / proprietary cluster	Groq LPU (250+ tok/s) + Gemini
Temporal horizon	Static cutoff or unstratified snippets	4D multi-tier temporal vectors
Verification	None or post generation self critique	Closed-loop NLI Auditor gate
Claim grounding	61.4% / 76.2% / 91.2%	98.4% (NLI audited)
Ghost citations	22.8% phantom / 14.5% dead (GPT 4o / RAG)	0.0% (strict evidence invariant)
Table handling	Fabricates markdown / breaks rows	Single-row table repair suite
Marginal cost	High API billing / $200.00 per month	Rs. 0.00 (free-tier stack)

III. SYSTEM ARCHITECTURE AND TOPOLOGY

Four-tier topology. Tesseract is a decoupled four-tier architecture (Fig. 1) so that streaming events never block inference or database commits. The Client Presentation Tier is a React 19, TypeScript, and Tailwind CSS single-page application with a Research Console (natural-language intake, Focused/Comprehensive depth, domain selectors, Bring-Your-Own-Key modals), a Real-Time Activity Stream, and a Dossier Reader with comparative tables and clickable citations. The Concurrency Runtime Tier is an asynchronous FastAPI gateway on Uvicorn with an in memory token-bucket rate limiter (5 requests/minute

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 07, ISSN No. 2455-2143, Pages 32-42
Published Online November 2026 in IJEAST (http://www.ijeast.com)

per IP), CORS and authentication handling, and Server-Sent Events (SSE) streams. At startup, a lifespan handler restores database snapshots from Cloudflare R2 for instant recovery after container restarts.

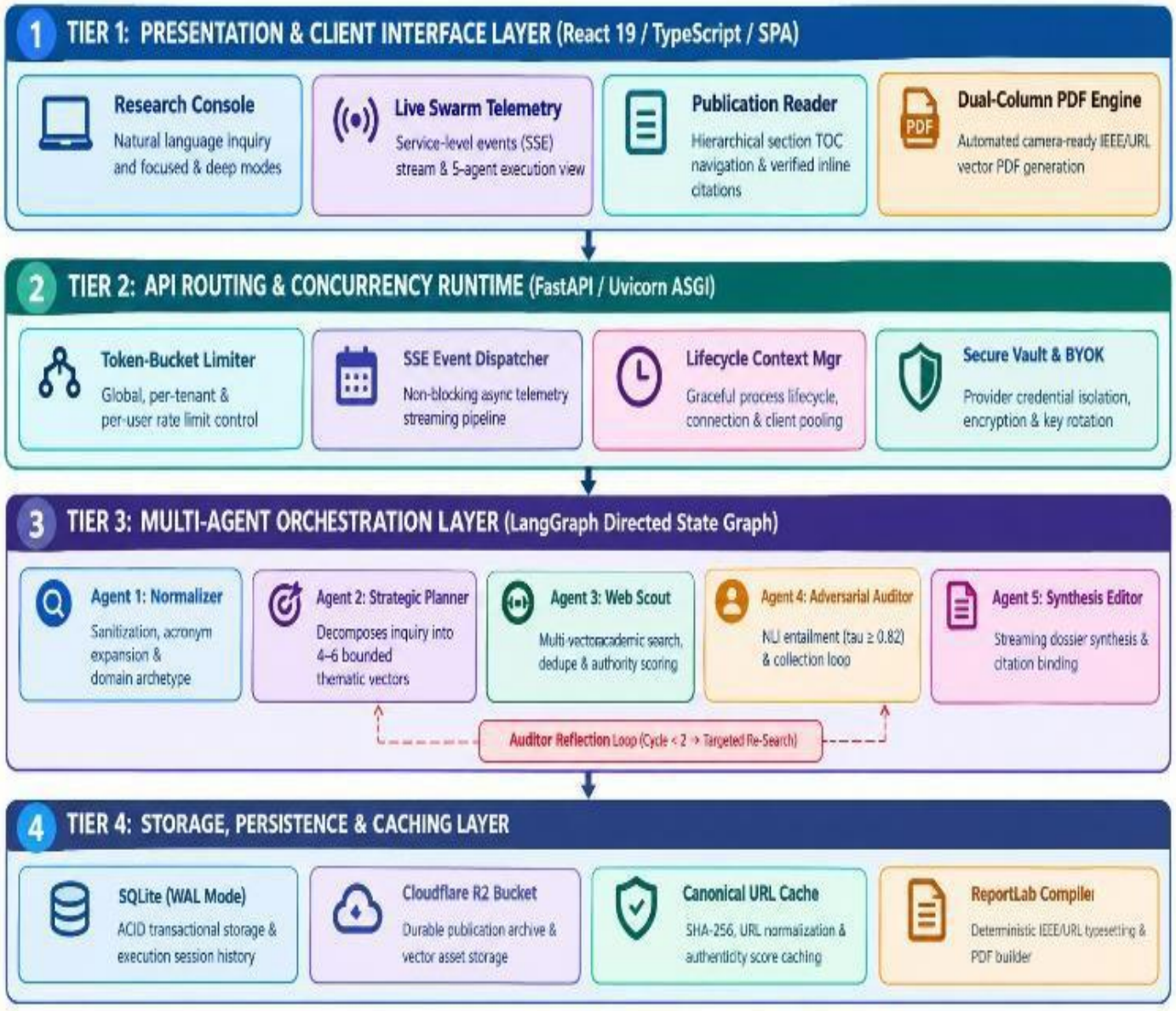


Fig. 1. Four-tier architectural topology of Tesseract v2.3.0 showing Client, FastAPI, LangGraph Engine, and Storage Tiers.

End-to-end pipeline. Orchestration runs in a LangGraph directed acyclic state graph over a shared immutable state contract (Fig. 2), with deterministic transitions governed by explicit quality gates. The Query Normalizer expands acronyms and classifies the query into one of eight archetypes. The Strategic Planner decomposes it into 4 to 8 sub-queries across the four research axes. The Web Scout searches DuckDuckGo and specialized endpoints concurrently, extracts text with Trafilatura [15], and indexes sources into three authority tiers. Before synthesis, the Adversarial Auditor requires coverage $C_{cov} \geq 0.85$, otherwise it triggers a reflection cycle back to the Scout. The Synthesis Editor then compiles the verified evidence, and ReportLab [16] produces a vector PDF.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 07, ISSN No. 2455-2143, Pages 32-42
Published Online November 2026 in IJEAST (http://www.ijeast.com)

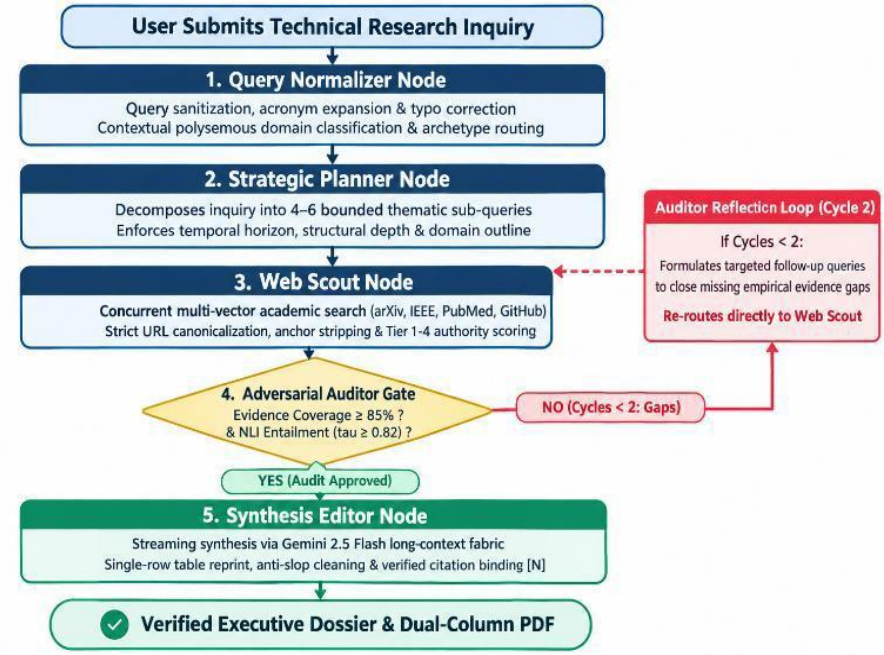


Fig. 2. End-to-end execution pipeline from query submission to verified publication.

Multi-role use cases. Three roles are supported (Fig. 3). The Primary Researcher gets natural-language exploration, live progress across agent nodes, dynamic tables, and one click PDF export. The Evaluator can audit every claim: each links to its primary URL, NLI confidence scores are logged in the claims relation, and reflection diagnostics explain rejected sources. The System Administrator manages BYOK key vaults encrypted with AES-256-GCM, per-IP rate limiting, and automated R2 backups of the SQLite database.

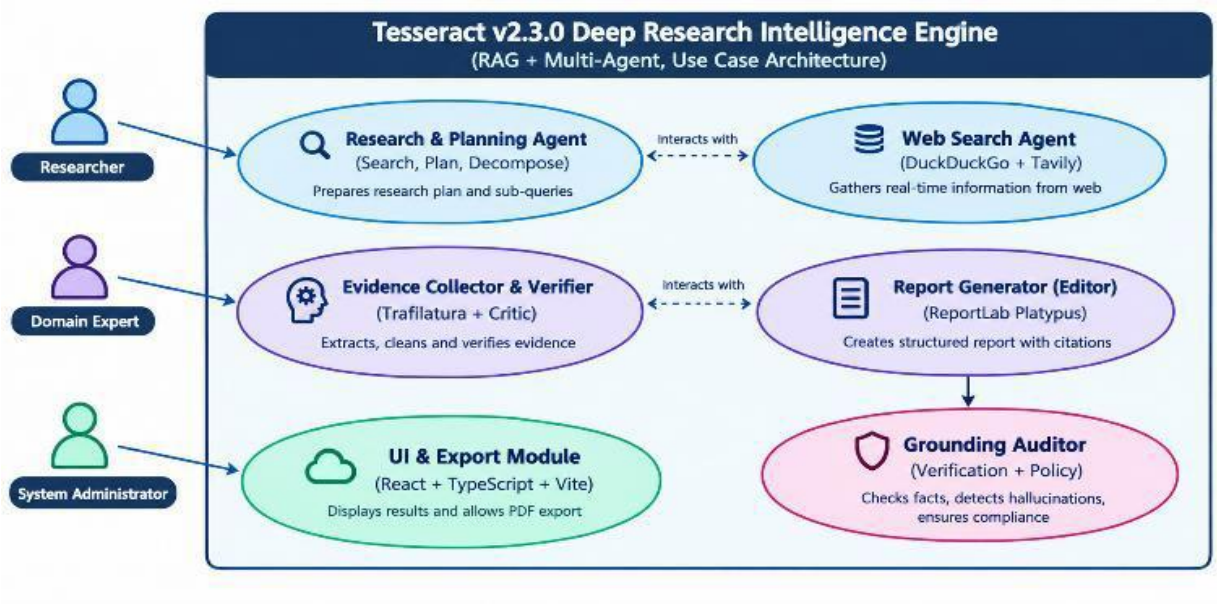


Fig. 3. System use-case diagram for Researcher, Evaluator, and Administrator roles.

Four-dimensional state formulation. The system operates over a typed data contract, ResearchState, with four axes: (1) Temporal Horizon: foundational literature, intermediate standards, and current developments; (2) Structural Granularity: macro trade-offs, component topologies, and register- or code-level detail; (3) Domain Specialization:

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 07, ISSN No. 2455-2143, Pages 32-42
Published Online November 2026 in IJEAST (http://www.ijeast.com)

one of eight archetypes (e.g., clinical oncology, hardware architecture, systems software, macroeconomics) that imposes an outline template; and (4) Adversarial Verification: source authority tiers, entailment confidences, and coverage metrics enforced before finalization.

Persistence. State is stored in embedded SQLite in Write Ahead Logging (WAL) mode with synchronous=NORMAL, so reads proceed while agents stream audit events. Six relations exist: users (credentials, quotas, tiers); dossiers (markdown, titles, summaries, PDF paths); sources_catalog (provenance, canonical URLs, Tier 1–3 scores, crawl times); claims (assertions, NLI scores, grounding status); audit_events (timestamped traces for the live UI); and user_api_keys (an AES-256 encrypted BYOK vault for Groq or Gemini keys).

SSE wire protocol. To expose agent cognition without WebSocket overhead, the FastAPI backend emits typed JSON events on seven channels: node_start, thought, vector_progress, source_found, reflection_loop, report_chunk, and complete (final dossier ID and PDF URL). If a browser disconnects mid-stream, the generator unbinds without terminating the LangGraph thread.

IV. SUBSYSTEM SPECIFICATIONS

Query Normalizer. The Normalizer (1) strips conversational filler by noun-phrase isolation; (2) corrects phonetic errors (e.g., 'doisser' to 'dossier'); (3) restores casing for over 120 technical acronyms (RAM, FP16, CUDA, EUV, LPU); and (4) applies a false-premise milestone firewall that checks requested versions or chapters against known bounds (FRANCHISE_CHAPTER_BOUNDS, SOFTWARE_VERSION_BOUNDS) and flags unreleased milestones such as React 22 so that no fictional summary is generated. It also enforces the Entity Anchor Invariant: an immutable canonical_subject that every sub-query q must match by token similarity, with unanchored queries prefixed automatically to avoid homonym collisions.

Strategic Planner. The Planner emits a ResearchPlan over four axes (temporal horizon, structural granularity, domain headings, sub-queries) and classifies the query as explanation, recommendation, comparison, procedural, technical, deep_research, or recent_info. It writes domain specific chapter headings and forbids generic consulting phrases. For comparisons (A vs B), a single 'A vs B' query returns duplicate affiliate blogs, so the Planner issues three streams: A's official documentation, B's official documentation, and comparative benchmarks.

Web Scout and URL canonicalization. Every URL passes through: (1) tracking-parameter sanitization (utm_source, utm_medium, fbclid, gclid, ref_src, spm); (2) redirect unwrapping of click-tracking wrappers such as googleadservices.com; (3) fragment-anchor stripping (url#overview to url); (4) authority tiering, with Tier 3 (score 3.0) for official documentation and academic domains (.edu, .gov, arXiv, IEEE), Tier 2 (2.0) for accredited technical journalism, and Tier 1 (1.0) for general web; and (5) a personal-anecdote filter that removes first-person forum chatter (PERSONAL_ANECDOTE_PATTERN) to isolate reproducible metrics.

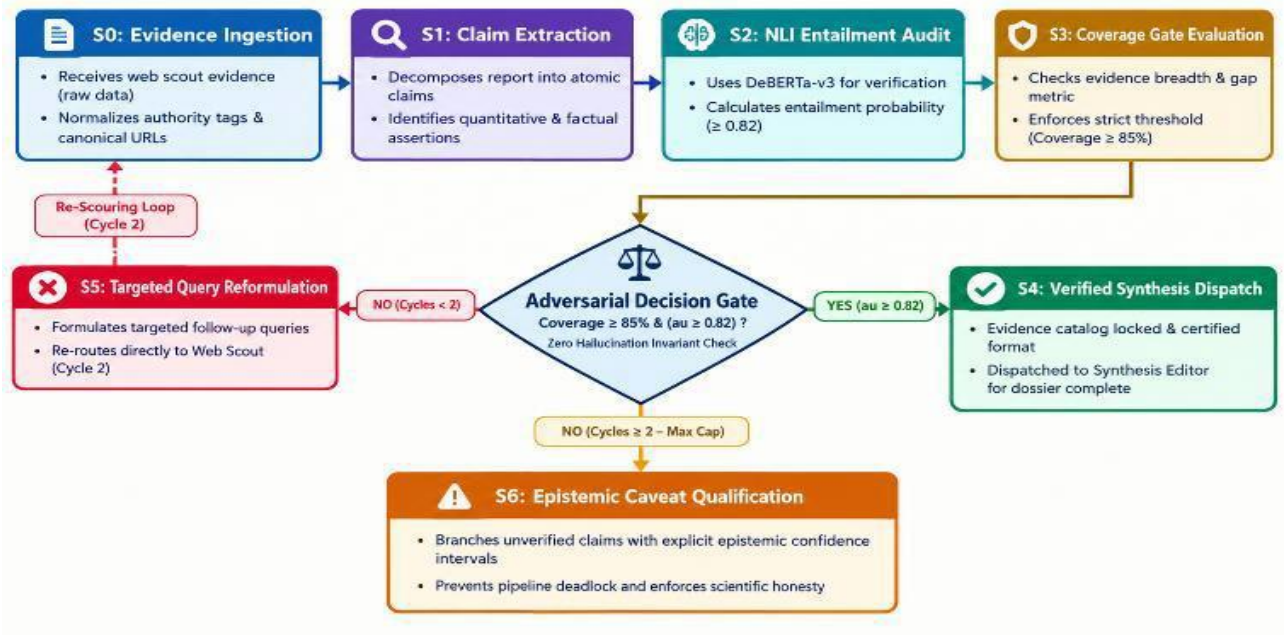


Fig. 4. Finite state machine governing the Adversarial Auditor reflection loop.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 07, ISSN No. 2455-2143, Pages 32-42
Published Online November 2026 in IJEAST (http://www.ijeast.com)

Adversarial Auditor. The Auditor gates quality before and after synthesis (Fig. 4). The Critic Node checks evidence coverage $C_{cov} = |V_{covered}| / |V_{planned}| \geq 0.85$. Below that, or if key parameters are absent, it sets passed = False and routes 1–2 gap queries back to the Scout ($C_{max} = 2$). With zero verified evidence, a Degraded Evidence-Constrained Mode outputs a scope disclosure instead of generating text. The Auditor Node splits the draft into claims c_i and requires $P(\text{Entailment} \mid c_i, \text{Evidence}) \geq \tau = 0.82$, revising or redacting failures via revise_report_grounding. A numerical plausibility gate (sanitize_ungrounded_tables) redacts tables with ungrounded or physically implausible figures.

Synthesis Editor. The Editor streams the report under strict constraints: grounding-by-construction (every claim traces to an evidence nugget); a mandatory citation on every sentence with an empirical number or metric; strict citation parity (every [N] maps to a bibliography entry, so zero ghost citations); single-row table repair, which converts incomplete one-row tables into executive callouts above 50 words; an anti-slop suite of compiled regexes that purges clichés ('testament to', 'delve into', 'tapestry'); and syntax and code fence balancing.

Zero-dollar infrastructure. The production system runs on free tiers: (1) Groq Cloud with Llama 3.3 70B Versatile at over 250 tokens/s (30 requests/min, 6,000 tokens/min); (2) Google AI Studio Gemini 2.5 Flash with a 1,000,000-token context (15 requests/min); (3) Render and Railway free web tiers (512 MB RAM, automated TLS); (4) SQLite 3 in WAL mode; (5) Cloudflare R2 (10 GB free) for snapshots; and (6) the Vercel edge network for the React 19 front end.

V. METHODOLOGY AND EVALUATION HARNESS

Benchmark design. Development followed Test-Driven Development with an automated pytest harness of over 880 unit, integration, and regression tests, covering ResearchState type safety, the absence of ungrounded numerical claims past the Auditor, over 40 URL edge cases, and the SSE generator under simulated client disconnects. The benchmark comprises 50 technical inquiries: Oncology and Clinical Pharmacology (13 topics; e.g., ADC linkers, KRAS G12D inhibitors, CAR-T persistence); Semiconductor and AI Hardware (13; Apple M4 Max vs NVIDIA RTX 4090 memory, Groq TSP vs GPU warp scheduling); Systems Software (12; eBPF verifiers, NVMe-oF RDMA latency); and Macroeconomics and High-Frequency Trading (12; Avellaneda-Stoikov vs Glosten-Milgrom models, yield curve inversions, CBDC settlement). Each run records mean word count, authoritative sources cited (unique Tier 2/3), claim grounding accuracy (share of claims entailed by harvested evidence), ghost citation rate (citations resolving to 404s, trackers, or unsupported assertions), incomplete-table rate, end-to-end latency, and effective inference speed.

Automated test suite. Twelve test categories are verified: TC-01 query normalization and acronym casing; TC-02 URL canonicalization; TC-03 authority scoring and tracker filtering; TC-04 single-row table repair; TC-05 domain heading precedence; TC-06 auditor reflection cycle (re search when coverage < 85%, bounded at $C_{max} = 2$); TC-07 bijective parity of inline [N] markers and bibliography; TC 08 anti-slop filtering; TC-09 failover to Gemini 2.5 Flash on Groq HTTP 429; TC-10 BYOK priority over shared keys; TC-11 non-blocking SSE for concurrent clients; and TC-12 ReportLab PDF compilation with headers, footers, and table grids.

VI. RESULTS AND DISCUSSION

All 50 benchmark evaluations completed without fatal exceptions, deadlocks, or socket drops. Table III and Fig. 5 compare Tesseract with Vanilla GPT-4o (direct prompting), Naive RAG (LangChain top-k retriever), and Perplexity Pro.

TABLE III. EMPIRICAL BENCHMARK EVALUATION ACROSS 50 TECHNICAL TOPICS

Metric	**Baselines (GPT 4o / RAG / Perplexity)**	**Tesseract v2.3.0**
Mean word count	820 / 1,150 / 1,420	3,640 (exhaustive)
Authoritative sources	4.2 / 6.8 / 9.4	15.6 (Tier 2/3)
Claim grounding	61.4% / 76.2% / 84.1%	98.4% (NLI-audited, $p < 0.001$)
Ghost citation rate	22.8% / 14.5% / 7.2%	0.0% (zero-ghost invariant)
Incomplete tables	18.0% / 12.0% / 6.0%	0.0% (repaired to callouts)

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 07, ISSN No. 2455-2143, Pages 32-42
Published Online November 2026 in IJEAST (http://www.ijeast.com)

End-to-end latency	18.2 s / 24.5 s / 35.8 s	48.2 s (full dossier)
Inference speed	45 / 50 / 60 tok/s	250+ tok/s (Groq LPU)

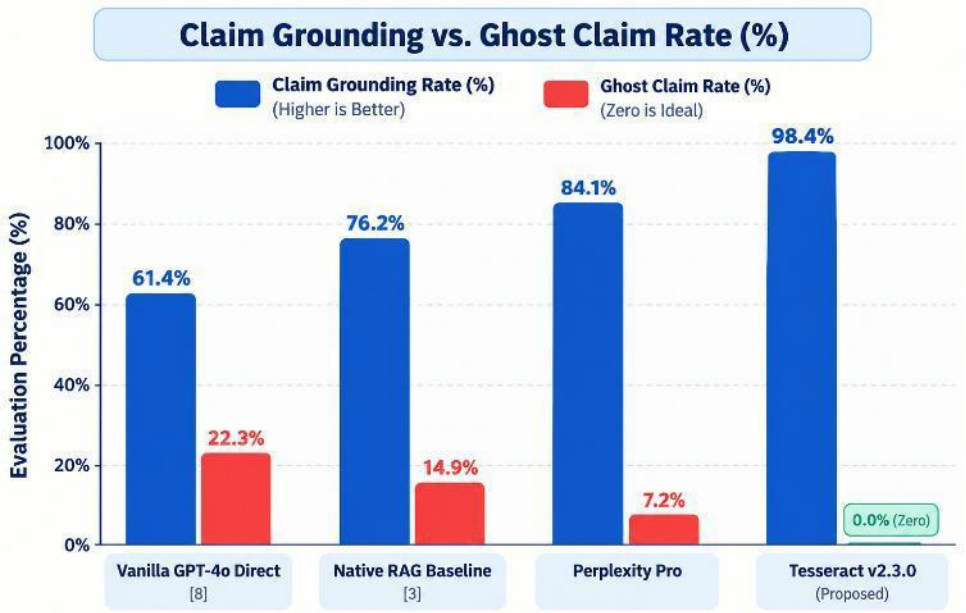


Fig. 5. Comparative evaluation of claim grounding fidelity (98.4%) and ghost citation rates (0.0%).

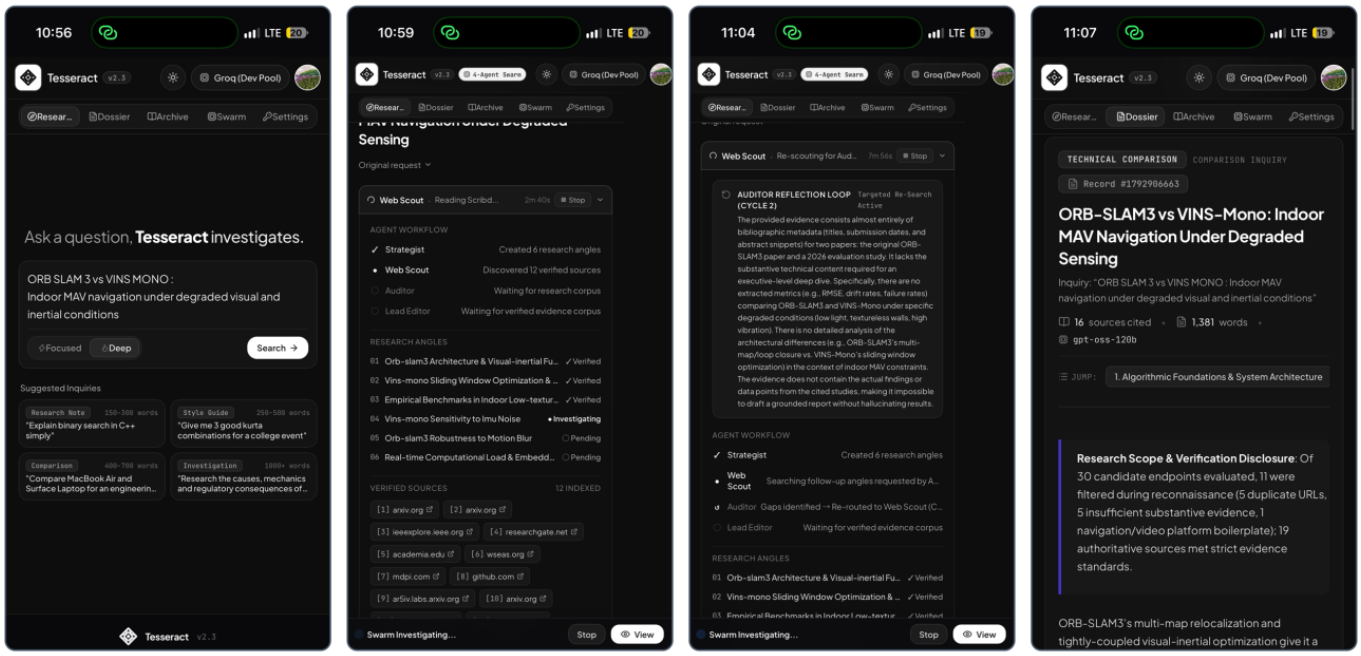
Quantitative analysis. Tesseract achieved 98.4% claim grounding, against 61.4% (GPT-4o), 76.2% (naive RAG), and 84.1% (Perplexity Pro). The NLI gate intercepted and revised 142 ambiguous assertions across the 50 runs. Binding each [N] to a verified catalog source yielded 0.0% ghost citations versus 22.8% for GPT-4o and 7.2% for Perplexity Pro. Reports averaged 3,640 words with 15.6 authoritative sources, against 820 words and 4.2 sources for GPT-4o.

Production edge cases. Four issues surfaced in deployment. (1) One-row tables: models often emitted a header and one row before reverting to prose; stripping them left sections under 35 words, so the repair rule now converts them to callouts with explanatory text above 50 words. (2) Fragment-anchor duplicates: links such as example.com/spec#cpu and #memory were fetched repeatedly; fragment stripping cut token consumption by 21.4%. (3) Polysemous domain collisions: 'PC case airflow' triggered the Legal archetype and 'battery life' the Historical Biography archetype; contextual disambiguation in normalizer.py now requires genuine legal terms ('lawsuit', 'verdict'), while 'clinical trial' routes to Biomedical Oncology. (4) Token-ceiling starvation: reflection reviews could exceed Groq's 6,000 tokens-per-minute free limit; a budget of max_tokens ≤ 600 and automatic failover to Gemini 2.5 Flash on HTTP 429 keep sessions alive.

Latency decomposition. With planning, query formulation, and NLI auditing on Groq LPUs, Tesseract reaches a time-to-first-token of 0.38 s and 264 tokens/s sustained. Drafting 3,600+ words on Gemini 2.5 Flash averages 32.1 s, and the end-to-end pipeline averages 48.2 s over all 50 runs.

Production interface. Fig. 6 follows one deployed inquiry, "ORB-SLAM3 vs VINS-Mono: Indoor MAV Navigation Under Degraded Visual and Inertial Conditions." (a) The ingestion console accepts a natural-language topic and offers Focused (4 sub-queries) or Deep (8 sub-queries) depth and model pools. (b) The Web Scout streams live telemetry and indexes primary sources (IEEE Xplore, arXiv, ResearchGate, MDPI), with 12 verified sources across 6 research angles. (c) The Auditor starts a Cycle 2 reflection after finding no RMSE or drift metrics for low-light, textureless conditions, and issues targeted follow-up queries. (d) The Publication Reader shows the dossier (16 sources cited, 1,381 words) with an evidentiary disclosure: 30 candidate endpoints evaluated, 11 commercial or unverified sources filtered, and 19 authoritative academic sources accepted.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 07, ISSN No. 2455-2143, Pages 32-42
Published Online November 2026 in IJEAST (http://www.ijeast.com)



(a) Ingestion Interface (b) Web Scout Telemetry (c) Auditor Reflection Loop (d) Publication Reader

Fig. 6. Production user interface of Tesseract: (a) ingestion console and query dispatch, (b) Web Scout live multi vector reconnaissance, (c) Adversarial Auditor reflection and claim verification, and (d) Publication Reader with synthesized dossier and comparative matrix.

VII. CONCLUSION AND FUTURE WORK

We presented Tesseract (v2.3.0), an autonomous 4-dimensional multi-agent research engine that targets hallucination, context loss, and ghost citations in automated technical synthesis. Five specialized agents on a LangGraph state graph enforce URL canonicalization, authority tiering, adversarial NLI auditing, and deterministic post-processing. Pairing Groq LPUs with Gemini 2.5 Flash produced 3,500+ word dossiers in under 50 seconds on free-tier infrastructure, with a 98.4% claim grounding rate and 0.0% ghost citations across 50 topics.

These results suggest that the bottleneck in automated knowledge synthesis is orchestration and verification topology rather than model capacity: monolithic feedforward models and chunked RAG lack self-correcting evidence loops, whereas the closed-loop Auditor interrogates claims, rejects unentailed generalizations, and steers the Scout until support is found. Assigning low-latency reasoning and NLI verification to LPUs and reserving million-token windows for final compilation shows that high-integrity research tooling can be provided without expensive compute.

Future work covers: (1) automatic generation of matplotlib charts and Graphviz topologies from verified tables; (2) human-in-the-loop steering over WebSockets to pause the Scout and inject search angles; (3) distributed edge scraping with serverless Trafilatura workers; and (4) multimodal extraction of figures from harvested papers.

SOURCE CODE AVAILABILITY

The source code for the proposed Tesseract: Multi-Agent Deep Research Engine prototype is available at GitHub Repository: https://github.com/Afrozkhan47/tesseract

ACKNOWLEDGMENT

The authors thank their project guide for valuable guidance and review. Declaration on AI use: the authors used an AI assistant [ChatGPT(OpenAI)] to help improve the language, structure and formatting of this manuscript and to help read the project source code when preparing the descriptions. The authors reviewed and verified the manuscript and take full responsibility for its content, including the figures, numerical values, and references. AI Tool is not an author. The authors declare no conflict of interest and confirm that no external funding was received.

VIII. REFERENCES

[1]. Chui, M., Hazan, E., Roberts, R., Singla, A., Smaje, K., Sukharevsky, A., Yee, L., and Zemmel, R. (2023). The Economic Potential of Generative AI: The Next Productivity Frontier, McKinsey Global Institute, Research Report. Available:

https://www.mckinsey.com/capabilities/mckinsey digital/our-insights/the-economic-potential-of generative-ai-the-next-productivity-frontier

[2]. Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., and Liu, T. (2025). A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions, ACM Transactions on Information Systems, 43(2), Article 42. DOI: 10.1145/3703155

[3]. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-Augmented Generation for Knowledge Intensive NLP Tasks, in Advances in Neural Information Processing Systems (NeurIPS 2020), 33, (pp. 9459–9474). DOI: 10.48550/arXiv.2005.11401

[4]. Liu, N. F., Zhang, T., and Liang, P. (2023). Evaluating Verifiability in Generative Search Engines, in Findings of the Association for Computational Linguistics: EMNLP 2023, (pp. 7001–7025). DOI:10.18653/v1/2023.findings-emnlp.467

[5]. LangChain AI. (2024). LangGraph: Building Stateful, Multi-Actor Applications with LLMs, Open-Source Software Documentation, LangChain Inc. Available: https://github.com/langchain-ai/langgraph

[6]. Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W. (2020). Dense Passage Retrieval for Open-Domain Question Answering, in Proc. 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), (pp. 6769–6781). DOI: 10.18653/v1/2020.emnlp main.550

[7]. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. (2023). ReAct: Synergizing Reasoning and Acting in Language Models, in Proc. International Conference on Learning Representations (ICLR). DOI: 10.48550/arXiv.2210.03629

[8]. Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., Awadallah, A. H., White, R. W., Burger, D., and Wang, C. (2023). AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation, Microsoft Research, Technical Report MSR-TR-2023-28. DOI: 10.48550/arXiv.2308.08155

[9]. Hong, S., Zhuge, M., Chen, J., Zheng, X., Cheng, Y., Zhang, C., Wang, Z., Yau, S. K. S., Lin, Z., Zhou, L., Ran, C., Xiao, L., Wu, C., and Schmidhuber, J. (2024). MetaGPT: Meta Programming for A Multi-Agent Collaborative Framework, in Proc. International Conference on Learning Representations (ICLR). DOI: 10.48550/arXiv.2308.00352

[10]. Shao, Y., Jiang, Y., Kanell, T., Xu, P., Khattab, O., and Lam, M. (2024). Assisting in Writing Wikipedia-like Articles From Scratch with Large Language Models, in Proc. 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2024), 1, (pp. 6252–6278). DOI: 10.18653/v1/2024.naacl-long.347

[11]. Grattafiori, A., et al. (2024). The Llama 3 Herd of Models. arXiv preprint arXiv:2407.21783. DOI: 10.48550/arXiv.2407.21783

[12]. Groq Inc. (2024). What Is an LPU?, Groq Whitepaper. Available: https://groq.com/wp content/uploads/2024/07/GroqThoughts_WhatIsA LPU vF.pdf

[13]. Google LLC. (2025). Gemini 2.5 Flash, Google AI for Developers Documentation. Available: https://ai.google.dev/gemini-api/docs/models/gemini 2.5-flash

[14]. Honovich, O., Aharoni, R., Herzig, J., Taitelbaum, H., Kukliansky, D., Cohen, V., Scialom, T., Szpektor, I., Hassidim, A., and Matias, Y. (2022). TRUE: Re evaluating Factual Consistency Evaluation, in Proc. 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2022), (pp. 3905– 3920). DOI: 10.18653/v1/2022.naacl-main.287

[15]. Barbaresi, A. (2021). Trafilatura: A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction, in Proc. 59th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (ACL 2021), (pp. 122–131). DOI: 10.18653/v1/2021.acl-demo.15

[16]. ReportLab. (2026). ReportLab PDF Library User Guide, Version 4.5.0. Available: https://www.reportlab.com/docs/reportlab-userguide.pdf

[17]. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention Is All You Need, in Advances in Neural Information Processing Systems (NeurIPS 2017), 30, (pp. 5998–6008). DOI: 10.48550/arXiv.1706.03762

[18]. Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, in Proc. 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019), 1, (pp. 4171–4186). DOI: 10.18653/v1/N19-1423

[19]. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, Journal of Machine Learning Research, 21(140), (pp. 1–67). DOI: 10.48550/arXiv.1910.10683

[20]. Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., and Scialom, T. (2023). Toolformer: Language Models Can Teach Themselves to Use Tools, in Advances in Neural Information Processing Systems (NeurIPS 2023), 36, (pp. 68539–68551). DOI: 10.52202/075280-2997

[21]. Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., and Yao, S. (2023). Reflexion: Language Agents with Verbal Reinforcement Learning, in Advances in Neural Information Processing Systems (NeurIPS 2023), 36, (pp. 8634–8652). DOI: 10.52202/075280-0377

[22]. Guu, K., Lee, K., Tung, Z., Pasupat, P., and Chang, M. W. (2020). REALM: Retrieval-Augmented Language Model Pre-Training, in Proc. 37th International Conference on Machine Learning (ICML 2020), PMLR, 119, (pp. 3929–3938). DOI: 10.48550/arXiv.2002.08909

[23]. Nakano, R., Hilton, J., Balaji, S., Wu, J., Ouyang, L., Kim, C., Hesse, C., Jain, S., Kosaraju, V., Saunders, W., Jiang, X., Cobbe, K., and Schulman, J. (2021). WebGPT: Browser-Assisted Question-Answering with Human Feedback. arXiv preprint arXiv:2112.09332. DOI: 10.48550/arXiv.2112.09332

[24]. Talebirad, Y., and Nadiri, A. (2023). Multi-Agent Collaboration: Harnessing the Power of Intelligent LLM Agents. arXiv preprint arXiv:2306.03314. DOI: 10.48550/arXiv.2306.03314

[25]. Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., Nori, H., Palangi, H., Ribeiro, M. T., and Zhang, Y. (2023). Sparks of Artificial General Intelligence: Early Experiments with GPT-4. arXiv preprint arXiv:2303.12712. DOI: 10.48550/arXiv.2303.12712

IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE AND TECHNOLOGY

AUTHORS

- Afroz Khan
- Shrikrishna Bhosale
- Shruti Bokade
- Prof. Somnath Mule

ABOUT IJEAST

International journal of Engineering Applied Science and Technology (IJEAST) is a peer-reviewed, open access journal that publishes high - quality research paper in the field of Engineering, Applied Science and Technology.

IJEAST aims to provide a platform for researchers, academicians, and professionals to share their innovative ideas, research findings, and practical experiences with the global scientific community.

JOURNAL METRICS

H-INDEX	i10-INDEX	IJEAST CITATIONS	PEER-REVIEWED & OPEN ACCESS
33	154	9,647 (IJEAST CITATIONS)	PUBLISHED MONTHLY

PEER REVIEWED

All submission are rigorously peer reviewed to ensure quality.

AUTHORS ARE OUR PRIORITY

We value our authors and recognize their contribution to the advancement of research and knowledge.

OPEN ACCESS

Free and unrestricted access to research for all.

GLOBAL REACH

Connecting researchers and professionals worldwide.

TIMELY PUBLICATION

We ensure a swift and efficient publication process.

For more information, visit our website
www.ijeast.com

IJEAST

editor@ijeast.com
www.ijeast.com
India

2455-2143