MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 3329-3333

Next-Generation Retrieval-Augmented Generation System: A Privacy-First Approach to Intelligent Document Analysis

Vigneshkumar S
Department of CSE
KGiSL Institute of Technology
Coimbatore, Tamilnadu, India
vigneshkumar_22csb56@kgkite.ac.in

Ramaranganathan S
Department of CSE
KGiSL Institute of Technology Coimbatore,
Tamilnadu, India
ramaranganathan_22csb22@kgkite.ac.in

Ridhun Krishnan R
Department of CSE
KGiSL Institute of Technology
Coimbatore, Tamilnadu, India
ridhunkrishnan_22csb25@kgkite.ac.in

Vishal Sarvani S Department of CSE
KGiSL Institute of Technology
Coimbatore, Tamilnadu, India
vishalsarvani_22csb60@kgkite.ac.in

Pradeep G
Assistant Professor, Dept. of CSE KGiSL Institute of Technology Coimbatore,
Tamilnadu, India pradeep.g@kgkite.ac.in

***Abstract*—One way to make big language tools more trustworthy is by pulling in outside facts when they answer. These setups usually lean on remote servers, which can be tricky if you care about who sees your information. Instead of guessing, some versions still get details wrong or miss pieces of longer texts. Handling different kinds of files - like images mixed with words - doesn't always work well either.**

This work introduces a web browser tool focused on privacy, built to analyze documents and answer questions smartly. All processing happens right inside your browser - nothing gets sent elsewhere. Instead of relying solely on one method, it uses both semantic matching and network-like data exploration together. By keeping everything local and blending these two techniques, results become more precise and context-aware.

Starting with mixed-format docs, it pulls out words, charts, formulas, or pictures straight from PDFs, Word files, spreadsheets, and scanned images. Instead of just trusting outputs, trust comes from triple-checking sources, matching facts across references, then confirming meaning through small-scale language tools that run on-site.

Testing across more than a thousand varied documents shows retrieval works 89 to 94 percent of the time, with answers arriving in about 2.3 seconds on average. Instead of relying solely on raw model output, this setup cuts made-up responses by nearly three-quarters. Privacy stays tightly controlled throughout every step. For businesses handling confidential files, legal teams digging through case history, or researchers pulling together study findings - this approach holds up under demand without exposing data. Performance remains strong even when workload increases.

***Index Terms*—Retrieval Augmented Generation, Knowledge Graphs, Privacy Preserving AI, Document Intelligence, Hallucination Detection, Local First Systems**

I. Introduction

A. *Background*

One big issue shows up when these systems face actual use outside labs. Think about how they handle facts under pres- sure. Machines like those built on GPT can grasp and produce human-like text pretty well. Yet problems start once real data must be correct every time. Tasks such as pulling summaries, shifting languages, responding to queries, or chatting still rely heavily on patterns seen before. What happens next often depends on prior examples rather than true reasoning. Even strong performance does not mean reliability in settings where accuracy matters most.

Wrong facts popping up out of nowhere? That's a big hurdle. When systems invent details that aren't true, things get shaky. Especially in areas like law or health records, accuracy matters more. Imagine reading a contract or diagnosis twisted by made-up lines. Finance reports full of fictional numbers wouldn't help anyone. Precision slips when fabrications sneak into analysis. Specific fields can't afford guesswork dressed as truth.

One way to tackle the problem? Pulling in outside info while building responses. Think of it like checking references before answering. These setups grab useful texts instead of depending only on what they already know. Information flows in first, then shaping happens around that base. Grounding replies becomes easier when fresh material guides the output. Still, nearly every current RAG setup runs through cloud systems, forcing people to send files to outside servers. That brings up serious issues around:

- Data privacy
- Regulatory compliance
- Intellectual property protection
- Enterprise data security

Working around these limits means building RAG systems that protect privacy by running only on the user's own device. While they avoid outside servers, such setups keep data close but demand efficient design. Because processing stays local, the system must handle tasks without relying on cloud resources. Still, keeping everything onsite introduces hurdles

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 3329-3333

in speed and storage. Even so, staying offline ensures sensitive information never leaves the machine.

B. Problem Statement

Even so, gains in RAG setups haven't fixed everything just yet - some issues still linger without answers:

1) **Privacy and Data Security:** Putting private files onto online servers often happens with AI tools that re- view documents. This move can expose information to breaches or unwanted eyes. Some systems store content far from user control, creating openings for misuse. Data might travel beyond secure boundaries when sent into remote networks.
2) **Hallucinated Responses:** Now here's a fact: large lan- guage models often invent answers that don't match the original texts. This habit makes them less dependable. People start doubting what they say. Accuracy slips when outputs float free from evidence. Trust fades just as quickly.
3) **Limited Context Handling:** Fewer words fit inside regular language model heads at one time.
4) **Poor Multi-Modal Processing:** Pages sometimes hold pictures, graphs, equations, plus grids - things old- style reading methods skip entirely. What slips through? Anything not plain words.
5) **Lack of Transparent Source Attribution:** It's tough for people to spot where answer parts came from in the documents. Sometimes the source bits stay hidden when responses appear. Finding which lines shaped replies often takes extra effort. You might struggle to match output chunks back to original sections. Tracing ideas to their roots inside files just isn't straightforward.

C. Research Objectives

This study aims to build a new kind of RAG system by tack- ling existing flaws - each goal shaping how it works. Instead of copying old models, fresh approaches guide its structure. With improvements rooted in practical testing, performance gains emerge step by step. Each part connects differently than before, avoiding past errors. Progress comes not from adding features but rethinking links between pieces:

- Develop a fully local privacy-preserving RAG architec- ture
- Reduce hallucinated responses through multi-layer veri- fication
- Support multi-modal document content processing
- Enable transparent source attribution
- Achieve high retrieval accuracy with low response latency

D. Research Contributions

The primary contributions of this work include:

1) **Privacy-First RAG Architecture:** Running inside your web browser, this tool handles file analysis while keep- ing everything on your device. Instead of sending in- formation out, it builds embeddings right where they're created. Processing happens in place, using only what's already present. No outside connection needed for any step along the way. Work stays private, contained within the local environment throughout.
2) **Hybrid Retrieval Mechanism:** Instead of relying on vectors alone, this system taps into connections be- tween ideas by blending embedding searches with paths through structured knowledge. Jumping from one related concept to another helps surface documents that actually match what matters. By weaving together numeric rep- resentations and logical relationships, it sharpens the ac- curacy of results without losing context. Relevance gets a boost when meaning is explored both mathematically and relationally.
3) **Multi-Modal Document Processing Pipeline:** Getting data out of PDFs works fine. Documents in Word format also get handled without trouble. Spreadsheets? They are included too. Even scanned images can give up their details here.
4) **Hallucination Detection Framework:** Confidence lev- els get assigned when unverified statements are flagged during a triple-step check. Step one spots what can't be backed up. Then step two weighs the evidence present. Finally, step three locks in how much trust each claim deserves.
5) **Dynamic Knowledge Graph Construction:** From piles of documents, key pieces emerge without help. One thing links to another through hidden threads. Patterns form where names meet actions. These ties show up clearly once spotted. Connections build quietly behind the scenes

II. Related Work

A. Retrieval-Augmented Generation

Starting off, Lewis and team rolled out the RAG system in 2020, linking document search directly into text-generating models. This method showed stronger results when handling NLP jobs that need deep factual recall. Still, behind the scenes, it depends a lot on central servers and online data hubs.

Starting off, Guu and colleagues introduced REALM in 2020 - a method that pulls in relevant info while training models, aiming to sharpen how context is grasped. Even though it works well, heavy-duty servers are needed behind the scenes.

What we do builds on earlier methods, letting users fetch and create data nearby without sharing private details.

B. Knowledge Graph Integration

Starting off differently, think of knowledge graphs as maps showing how things connect. Work by Bordes and team back in 2013 showed these map-like structures, when turned into number patterns, boost performance on question-response tools. Though not obvious at first glance, turning links between items into math forms helps machines understand better. Instead of just listing facts, such methods let systems infer answers they've never seen before. While many approaches

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 3329-3333

exist, this one stands out due to its precision in capturing meaning through structure.

Not stuck like old databases, this setup builds a live web of info pulled straight from documents, so answers fit the situation better.

C. *Hallucination Detection in LLMs*

A fresh look at made-up answers in computer language came from Ji and team last year. Because false details pop up so easily, linking facts to real sources helps. Checking claims after they're generated also cuts down fiction. One thing clear: trust needs backup.

A fresh look at spotting false outputs begins by tracing where info comes from, then flows into checking if meanings hold together across parts. One step ties back to original references while another watches how ideas connect logically throughout. Instead of guessing, it builds confidence through cross-matching facts and tracking coherence patterns. Each stage acts like a filter - first grounding claims in evidence, next ensuring they make sense when linked.

D. *Privacy-Preserving Artificial Intelligence*

Fed up with privacy leaks, some turn to federated methods - seen in recent work by Xu and team (2021). Still, behind the scenes, scattered servers keep holding the reins. Focusing locally first, this design keeps data under user control, protect- ing privacy fully. Unlike static knowledge bases, the proposed system dynamically constructs a document-derived knowledge graph, enabling context-aware retrieval.

III. System Architecture

A setup built on layers splits into five main parts. Each piece fits together, working step by step. One feeds into the next without overlap. Design keeps things separate but linked. Every section handles its own task clearly:

1) User Interface Layer
2) Document Processing Pipeline
3) Core Retrieval Engine
4) Verification and Quality Control
5) Local Storage Layer

Built right into your browser, it runs without outside help by handling tasks where you see them. Instead of sending data away, everything stays put through built-in memory tools that keep things running on their own.

A. *Document Processing Pipeline*

Files of various kinds can be brought into the system - PDFs, Word docs, spreadsheets make it possible through the intake component. Different types land without issue when processed here:

- PDF documents
- Microsoft Word files (DOCX)
- Excel spreadsheets (XLSX, CSV)
- Image files
- Plain text documents

Starting off, text gets pulled by tools like PDF.js, Mammoth.js, or SheetJS instead of manual copying. Images go through OCR methods so words inside them become readable. Pieces of the pulled content are split into parts that overlap, helping make embeddings faster to create plus easier to find later.

B. *Embedding Generation*

A fresh twist on text pieces: they turn into tight number bundles through smart sentence tools. These numerical forms pack meaning without losing shape. Sentence transformers handle the shift smoothly, almost like translation but for machines. Each chunk gets its own coded form, ready for deeper tasks later. The process runs quietly behind scenes, turning words into silent math versions.

Model Used: all-MiniLM-L6-v2

Key characteristics:

- 384-dimensional embeddings
- Low computational overhead
- Ready for close meaning matches

Right inside your browser, embeddings take shape through ONNX Runtime running on WebAssembly - nothing ever leaves your device. Data stays put, processed entirely on site without reaching outside servers. The whole operation runs silent, no signals sent, just local computation doing its work. Your information never travels, shielded by design from any network exchange.

C. *Knowledge Graph Construction*

A web of ideas takes shape as pieces pulled from texts link up. Moving parts shift while connections form between what was found. Items once separate now relate through changing paths. Structure grows where meaning meets arrangement. What emerges is never fixed but always adjusting.

Node Types:

- Documents
- Entities
- Concepts

Edge Types:

- Mentions
- Related-To
- Part-Of
- Derived-From

With this graph setup, thinking through relationships becomes possible - accuracy when pulling data on tough questions gets a boost. A web of links shapes how details connect, making searches smarter without extra effort.

D. *Hybrid Retrieval Strategy*

One way pulls data on the fly. Another waits till needed. A third mixes both, depending on what happens. The system picks one without warning:

1) Vector Similarity Retrieval
2) Knowledge Graph Traversal
3) Hybrid Retrieval (combined results)

Falling somewhere between two systems, the blend sorts out- comes by score adjustments - outperforming either standalone technique when tested side by side.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 3329-3333

E. Hallucination Detection and Verification

A single answer goes through three checks before it shows up. First, it gets reviewed for clear mistakes. Then, someone else looks at how accurate it is. Finally, a third check makes sure nothing was missed along the way:

- **Stage 1 - Source Check:** A single source backs each statement made in the output. Where a piece comes from stays clear through matching parts pulled earlier. Nothing appears without its match showing up first. Tracing where ideas begin matters deeply here. Every line ties directly to something found before.
- **Stage 2 - Factual Consistency Check:** Finding mis- matches by checking new claims against original texts instead of assuming accuracy.
- **Stage 3 - Detecting False Perceptions:** A single thought builds on another when a homegrown system checks if replies match the original text. What matters here is how meaning stays steady across sentences formed by machine output. Matching ideas becomes clear through quiet comparisons made inside small software minds.

One way to get the final confidence number involves weighing different test results differently. How much each result counts depends on its importance in the overall check.

IV. Implementation

Running locally, the setup uses current web methods to- gether with artificial intelligence processed on device. Though built recently, it relies on tools that work without cloud support.

Frontend:

- React.js
- Tailwind CSS
- Framer Motion

AI Components:

- Ollama (local LLM inference)
- HuggingFace Transformers
- ONNX Runtime
- Tesseract OCR

Document Processing:

- PDF.js
- Mammoth.js
- SheetJS

Storage:

- IndexedDB (local browser database)

Documents live in one of nine spots meant for storage. Em- beddings sit separate, each tucked into its own space. Entities appear across several locations built just for them. Queries move through dedicated sections set aside early on. System metrics gather quietly inside assigned holders. Relationships link up within distinct containers made to fit. Each store handles a single job without overlap. Storage paths stay split by design. The whole setup runs using exactly nine parts.

V. Experimental Evaluation

A. Dataset

One test used a collection of texts, including these items:

- 1,000 documents
- Multiple formats (PDF, DOCX, XLSX, Images)
- Total dataset size: 2.3 GB

Out of thin air came fifty questions each, stacked into four buckets. These added up to five hundred total tests put together on purpose:

- Factual queries
- Analytical queries
- Multi-hop reasoning queries
- Summarization queries

B. Retrieval Performance

A mix of search methods worked much better than relying on just one. Despite common assumptions, combining tech- niques showed clearer results.

TABLE I
Retrieval Performance Comparison

Method	Top-1 Accuracy	Top-5 Accuracy
Vector Retrieval	67.2%	82.4%
Graph Retrieval	71.8%	85.6%
Hybrid Retrieval	78.4%	91.2%

C. Response Quality

Table II outlines the performance metrics in terms of hallucination rates and accuracy.

TABLE II
Response Quality Metrics

Metric	Baseline LLM	Proposed System
Hallucination Rate	31.2%	8.3%
Source Attribution	0%	94.2%
Factual Accuracy	68.9%	91.7%
User Satisfaction	3.2 / 5	4.6 / 5

D. Performance Benchmark

Achieving 2.3 seconds on average, the system handled questions tied to documents from start to finish. Though built for accuracy, speed came through each step without delay. Response times stayed close to that mark, mainly because processing flowed directly from input to answer. Because every phase linked tightly, lag never piled up along the way. Fewer than 200 megabytes stayed in use, so most desktops can handle it just fine.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 3329-3333

VI. Applications

The proposed system has several real-world applications:

- **Enterprise Knowledge Management:** Finding files in- side a company stays safe, yet private details never leave protection. While searching happens, access re- mains locked down tight.
- **Legal Document Analysis:** Firms that work with law now spot contract parts faster using smart tools. Case pa- pers get sorted without manual checks, thanks to pattern recognition systems.
- **Academic Research:** Researchers can synthesize find- ings from multiple research papers.
- **Healthcare Record Analysis:** Folks in medicine study health files without breaking privacy rules.
- **Financial Intelligence:** Finding meaning in numbers of- ten begins with a closer look at company records instead of just glancing over totals. Pages filled with figures start telling stories once someone digs beneath the surface.

VII. Limitations and Future Work

Even so, the suggested setup works well but still has flaws. Yet gaps linger despite solid results shown by the model.

Current limitations include:

- Browser storage capacity constraints
- Only basic help available when working with tough math formulas
- Dependence on local LLM infrastructure

Focusing ahead lands on these points:

- GPU acceleration via WebGPU
- Distributed storage integration
- Multilingual support
- Advanced knowledge graph visualization
- Mobile-compatible progressive web applications

VIII. Conclusion

A fresh take on smart document review emerged here - built around strong privacy, using retrieval-augmented generation. The method keeps data safe while digging into documents in- telligently. Privacy shaped every layer of the design. Instead of pulling answers openly, it draws insights quietly behind secure walls. Document understanding improved without trading off protection. Each step worked to balance access with caution. Starting off strong, the model pulls together mixed search methods, logical connections from data networks, while han- dling various file types at once - leading to sharper results and steadier answers. What stands out is how each piece supports better precision without slowing things down.

Testing shows the new design cuts false answers by more than seven out of ten. What stands out is how it still gets facts right nearly every time. Speed stays quick too - under three seconds per reply - not a moment longer.

A fresh approach takes shape when data stays private but still fuels smart systems. Where companies handle internal knowledge, it fits without fuss. Legal teams digging into past cases see benefits too. Hospitals study patient patterns safely under this design. Even scholars testing new ideas find room to move. Each field gains ground in its own way.

References

[1] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *NeurIPS*, 2020.
[2] K. Guu et al., "REALM: Retrieval-Augmented Language Model Pre-Training," *ICML*, 2020.
[3] A. Bordes et al., "Translating Embeddings for Modeling Multi-Relational Data," *NeurIPS*, 2013.
[4] Z. Ji et al., "Survey of Hallucination in Natural Language Generation," *ACM Computing Surveys*, 2023.
[5] R. Xu et al., "Federated Learning for Privacy-Preserving Artificial Intelligence," *IEEE Security & Privacy*, 2021.
[6] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks," *EMNLP*, 2019.
[7] A. Vaswani et al., "Attention Is All You Need," *NeurIPS*, 2017.
[8] J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transform-ers," *NAACL*, 2019.
[9] T. Brown et al., "Language Models are Few-Shot Learners," *NeurIPS*, 2020.
[10] C. Raffel et al., "Exploring the Limits of Transfer Learning with T5," *Journal of Machine Learning Research*, 2020.