Le RAG n'a (presque) rien inventé

7 July 2026 • 19 min de lecture

RAG Information Retrieval

Ce texte va sans doute ressembler à une discussion de petits vieux sur le banc d'un village du sud de la France avec le bruit des cigales, un pastis à la main et un léger fond de nostalgie. La discussion commencerait ainsi : RAG, recherche agentique ou "deep research", peu importe le nom; dans tous les cas, il faut faire du *retrieval*. Ce que l'industrie a présenté comme une invention de 2023 repose, pour l'essentiel, sur une discipline cinquantenaire : la recherche d'information. Son intégration aux LLM est une évolution architecturale réelle mais pas une génération spontanée. Ce qu'elle a de neuf tient en une boucle : raisonner, chercher, raisonner encore. Ce qu'elle coûte tient en un troc mesurable : de la lecture précise des sources contre une reconstruction fluide qui se trompe encore trop souvent. L'adversaire de ce texte n'est pas la recherche, mais la vulgate qui a vendu une rupture née de

rien et la pratique qui l'a implémentée. Les équipes de Google, Microsoft, Anthropic ou OpenAI connaissent BM25, TREC, les facettes et les graphes mieux que quiconque ; pourtant, une couche discursive s'est interposée entre ces équipes et le reste du monde, qui implémente la version racontée dans les pitchs. L'avantage de nos papis sur le banc, c'est qu'ils ont toute l'histoire.

Ce qui est acquis

Au sommet du marché, les meilleurs systèmes exécutent déjà la chaîne complète : récupérer, raisonner, générer, citer, laisser naviguer vers les sources. Par exemple, NotebookLM de google ancre chaque affirmation à un passage cliquable, les produits de deep research rendent des rapports sourcés, les API de citations existent chez tous les grands fournisseurs. Une partie du problème tient davantage aux implémentations médianes (ni recherche, ni grand éditeur) qu'au paradigme lui-même.

Alors où est le problème résiduel ? Dans deux écarts, tous deux mesurés. L'écart médian d'abord : ce ne sont pas les laboratoires qui déploient la majorité des systèmes, ce sont des équipes pressées qui implémentent le pipeline naïf et selon les retours de terrain, près de 40 % des échecs se produisent silencieusement (https://lushbinary.com/blog/rag-retrieval-augmented-generation-production-guide/) à l'étape de récupération. L'écart comportemental ensuite : la "navigation" existe dans l'architecture mais n'est pas empruntée dans l'usage. Les chatbots renvoient vers les sources un trafic inférieur de 96 % à celui de la recherche classique (données TollBit : https://www.niemanlab.org/2025/03/ai-search-engines-fail-to-produce-accurate-citations-in-over-60-of-tests-according-to-new-tow-center-study/) ; testés sur l'attribution, les produits phares eux-mêmes échouent plus de six fois sur dix, avec un aplomb constant (étude du Tow Center, Columbia, huit moteurs génératifs, 200 tests : https://www.cjr.org/tow_center/we-compared-eight-ai-search-engines-theyre-all-bad-at-citing-news.php) ; et en recherche juridique, les outils ancrés hallucinent encore sur 17 à 33 % des requêtes [3].

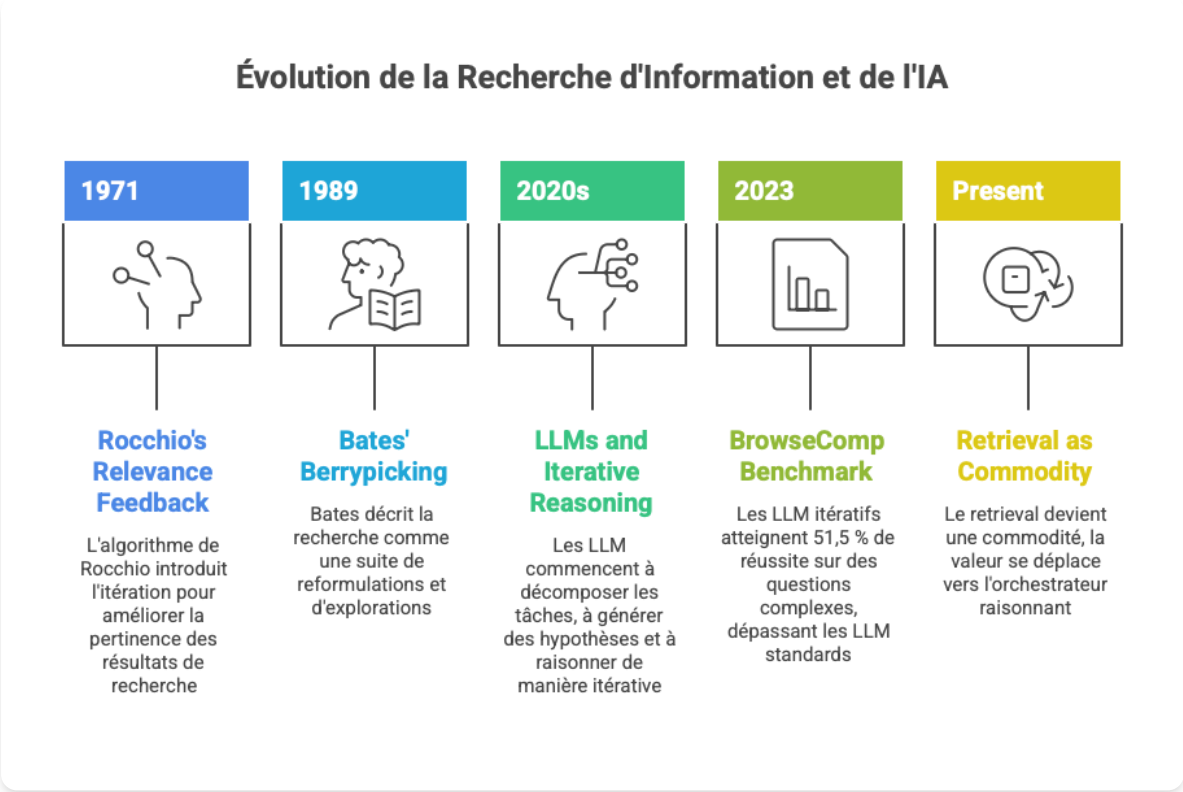
Il n'en reste pas moins que les systèmes augmentés fonctionne "mieux" que les modèles nu (ne serait-ce que pour la fraicheur des données). La confiance dans ces systèmes ne se jouera ni dans le choix de l'index ni dans la taille de

la fenêtre, mais dans la fermeture de ces deux écarts : que la pratique médiane rejoigne la frontière, et que vérifier coûte moins cher que croire.

La rupture véritable et son héritage

Reconnaître l'héritage n'oblige pas à nier la nouveauté ; encore faut-il la situer au bon endroit. Elle n'est ni dans l'embedding ni dans la base vectorielle : elle est dans la boucle de raisonnement. L'IR avait théorisé l'itération depuis un demi-siècle (le retour de pertinence de Rocchio date de 1971 [14], le « berrypicking » de Bates décrit dès 1989 la recherche comme une suite de reformulations [15]) mais toujours pour un humain aux commandes. Ce qu'aucun moteur ne savait faire, les LLM et leurs ecosystèmes le font désormais seuls : décomposer une tâche, générer des hypothèses, changer d'angle, raisonner sur les résultats et décider de chercher encore. La machine exécute le comportement que la discipline (IR) ne pouvait qu'outiller. L'écart de performance peut se mesurer sur BrowseComp (un benchmark de 1 266 questions conçues pour être introuvables en une seule requête) : les LLM standards restent typiquement sous 10 % tandis qu'OpenAI Deep Research, dont l'architecture repose précisément sur cette boucle itérative, atteint 51,5 % [13]. Ce n'est pas la rupture conceptuelle que ce chiffre mesure, c'est son effet sur un cas précis ; il donne toutefois la bonne intuition de l'ordre de grandeur de ce que "l'orchestration raisonnante" apporte au-delà du retrieval seul.

La conséquence structurante : si la rupture est la boucle, la valeur est dans ce qui la pilote (l'orchestrateur qui raisonne) et non dans les outils qu'il appelle. Le retrieval n'est plus un composant différenciateur ; il est en train de devenir une commodité sur laquelle tout repose et que personne ne vendra. Ce glissement, visible dans tous les signaux qui suivent, commande l'architecture des systèmes et la politique des investissements.



Les signaux de mi-2025 à mi-2026

Dans les systèmes réels de 2026, la recherche lexical, vectorielle, agents, graphes et mémoire coexistent, la récupération hybride est devenue le choix par défaut.

La recherche agentique. Claude Code a abandonné son RAG vectoriel initial (index périmé dès qu'une fonction fraîchement écrite n'est pas indexée, problèmes de permissions et de confidentialité) pour du glob/grep piloté par le modèle, qui a tout battu ; l'outillage a suivi (Cursor, Windsurf, Devin). Un papier d'Amazon à AAAI 2026 chiffre le phénomène : un agent limité aux mots-clés atteint plus de 90 % des métriques d'un RAG vectoriel en question-réponse documentaire [8]; résultat (pas un théorème) : l'hybride reste le défaut sur les corpus non structurés, et le grep itératif brûle des tokens là où un index dense répondrait en un seul passage. La nouveauté n'est pas que la machine cherche, c'est qu'elle raisonne entre deux recherches, reformule, choisit un nouvel angle : le comportement de Rocchio [14], automatisé. Et nommer l'ironie : appeler « agentique » ce qui est, au fond, de la recherche d'information pilotée par un agent.

Le context rot. Le terme a été forgé par Chroma Research en 2025 pour désigner un phénomène mesurable : la dégradation continue de la précision d'un LLM à mesure que son contexte s'allonge, bien avant que la fenêtre ne soit pleine [19]. Ce n'est pas un dépassement de capacité mais un effritement silencieux de la qualité. Un modèle annoncé à 200 000 tokens peut montrer des pertes significatives à 50 000 tokens d'entrée ; un modèle à un million de tokens ne raisonne pas fiablement sur un million de tokens. Trois mécanismes s'accumulent : l'effet « lost in the middle » documenté par Liu et al. (Stanford, TACL 2024), le modèle traite bien les extrémités du contexte mais perd le fil de ce qui est au centre ; la dégradation amplifiée quand la requête est sémantiquement distante des passages (précisément le cas réaliste) ; et l'interférence entre passages non pertinents qui s'accumulent. Chroma a mesuré cela sur dix-huit modèles frontières, tous sans exception. Un conflit d'intérêts à mentionner : Chroma vend une base vectorielle (une dégradation sévère du contexte long sert directement leur argument commercial), ce que plusieurs commentateurs ont noté. Les résultats restent cohérents avec la littérature antérieure sur le « lost in the middle » et méritent d'être pris au sérieux, avec cette réserve. Le consensus formé début 2026 est un hybride : récupérer 50 à 200 mille tokens de preuves, raisonner longuement dessus, router requête par requête entre les deux modes selon la nature de la question. L'économie confirme : une seconde de latence côté récupération contre des dizaines côté prompt saturé. La question binaire « RAG ou contexte long ? » est morte ; reste un budget de tokens à arbitrer.

GraphRAG, sans les gants. Le papier fondateur de Microsoft Research (Edge et al., avril 2024 [20]) part d'un constat simple : le RAG vectoriel classique échoue sur les questions qui portent sur l'ensemble d'un corpus « quels sont les grands thèmes de cette collection ? », « quels risques traversent tous nos contrats ? ». Pour une question de ce type, récupérer les k passages les plus similaires ne suffit pas : il faut agréger. GraphRAG résout le problème en deux temps. D'abord, hors-ligne, un LLM extrait les entités et relations du corpus pour construire un graphe de connaissances, puis applique un algorithme de détection de communautés pour regrouper les entités proches et générer des résumés de communautés pré-calculés. Ensuite, à la requête, ces résumés

sont interrogés pour les questions globales, tandis que le graphe local (entités, relations, passages adjacents) sert les questions factuelles précises.

Maintenant, le coût et c'est là que les présentations de conférences deviennent vagues. L'indexation initiale de GraphRAG a longtemps été prohibitive : en début 2024, indexer un corpus juridique de 5 Go coûtait jusqu'à 33 000 dollars en appels LLM, chaque entité et chaque résumé de communauté nécessitant un traitement LLM pendant la phase hors-ligne. LazyGraphRAG (Microsoft Research, novembre 2024) a résolu ce problème en différant la summarisation des communautés à l'heure de la requête : le coût d'indexation descend à 0,1 % de GraphRAG complet, comparable à un RAG vectoriel. Mais le coût d'indexation n'est pas le seul mur. Des travaux récents [21] mesurent une latence à l'inférence 2 à 3 fois plus élevée que le RAG vectoriel, et un index qui croît de façon super-linéaire avec la taille du corpus, compliquant les mises à jour incrémentales. Et ces chiffres ne comptent pas ce qui échappe aux benchmarks : la résolution d'entités et l'*entity linking* à l'échelle (le même client sous quatre graphies, la même entreprise sous deux noms), la qualité des liens qui conditionne tout l'aval du raisonnement, la maintenance quand le corpus évolue, la dérive du schéma au fil des versions, les reconstructions périodiques. Beaucoup de pilotes GraphRAG meurent en maintenance, pas en démo.

Ce qui le sauve tient en deux arguments solides. D'abord, l'unicité fonctionnelle : pour les questions thématiques sur un corpus entier, il n'existe pas d'alternative connue qui passe à l'échelle, aucun top-k vectoriel ne peut synthétiser des milliers de documents en une seule passe.

La mémoire. Le RAG répond à « quels documents ressemblent à cette requête ? » ; les agents de 2026 demandent « que sais-je de cet utilisateur, et qu'est-ce qui a changé depuis hier ? ». Mem0 [12] traite les faits comme une base de données (ajouter, mettre à jour, supprimer) : un déménagement remplace l'ancienne adresse au lieu de l'ensevelir sous un chunk contradictoire, opération structurellement impossible dans un sac de vecteurs. Zep[22] ajoute la temporalité par un graphe de connaissances où chaque fait porte sa période de validité. La direction est nette : la récupération

devient un cas particulier de la gestion d'état, et l'état et la temporalité sont les angles morts structurels du pipeline naïf.

Les cartouches sémantiques / *Cartridges*. Cartridges [7] entraîne hors-ligne un petit cache d'attention par corpus par auto-étude (le modèle génère des conversations synthétiques sur les documents et se distille dedans) puis le charge à l'inférence : 38,6 fois moins de mémoire, 26,4 fois plus de débit, cartouches composables sans réentraînement. Engram, fondée par les auteurs, sort de furtivité en juin 2026 avec 98 millions de dollars et Microsoft, Notion et Harvey en partenaires. L'argument économique est brutal : un contrat de 70 000 mots gonfle en cache KV à plus de 100 gigaoctets (une inflation d'environ 250 000 fois) reconstruits à chaque requête. La cartouche pré-paie la lecture une fois et l'amortit sur toutes les requêtes suivantes. La leçon : après le pipeline, le comportement et l'état, voici la connaissance compilée (redoutable sur les coûts, muette sur la provenance) : une cartouche ne cite pas ses pages.

La surface d'attaque élargie. Une précision s'impose avant tout le reste : l'empoisonnement de corpus n'est pas une vulnérabilité propre au RAG. Injecter des documents forgés pour manipuler les résultats, c'est le spamdexing des années 2000 qui préexiste aux embeddings, aux vecteurs et aux LLM. Ce qui a changé n'est pas la surface, c'est la dangerosité : autrefois, le document empoisonné finissait dans un classement parmi des millions, et l'utilisateur le lisait en jugeant sa crédibilité. Aujourd'hui, il finit dans le contexte d'un LLM qui lui fait confiance par construction, le synthétise sans le montrer, et peut agir sur sa base via des outils.

PoisonedRAG [11] a quantifié l'amplification : cinq documents forgés suffisent à dicter environ 90 % des réponses ciblées dans des bases de millions d'entrées. Branché sur des agents outillés via MCP, le document piégé ne fait plus seulement mentir, il fait agir. La défense qui marche est architecturale : provenance signée, contrôle des flux, aucun agent de synthèse au contact de l'évidence brute non fiable [11]. Le signal est réel ; mais son honnêteté intellectuelle exige de dire que le RAG en est l'amplificateur, pas l'inventeur.

La mécanique, pour les praticiens

L'architecture haut niveau (récupérer, raisonner, citer) donne l'illusion d'être simple. C'est dans les mécanismes intermédiaires que les systèmes divergent, que les démos s'effondrent en production, et que l'héritage de la recherche d'information se voit le plus clairement.

Tout commence avant même la récupération, dans la façon dont la requête est préparée. Une requête brute de l'utilisateur est rarement la meilleure entrée pour un index : elle est ambiguë, trop courte, formulée dans le registre de quelqu'un qui cherche plutôt que dans celui des documents qui répondent. La réécriture et l'expansion de requêtes (reformuler, enrichir de synonymes, générer des formulations hypothétiques proches du style des sources cibles) est l'héritière directe du retour de pertinence de Rocchio [14] : l'idée que la requête doit s'affiner à mesure qu'on sait ce qu'on cherche. Les agents vont plus loin : ils décomposent une question complexe en sous-requêtes indépendantes, planifient quelles sources interroger dans quel ordre, et estiment le budget de tokens à consacrer avant d'exécuter. Ce travail en amont est souvent ce qui sépare un agent qui répond bien d'un agent qui répond vite.

Au moment du rappel lui-même, la recherche hybride lexicale et vectorielle est devenue le défaut recommandé. La recherche lexicale (BM25) excelle sur les termes rares, les noms propres et les identifiants exacts ; la recherche vectorielle par embeddings capture la proximité sémantique quand les mots diffèrent mais le sens converge. Aucune des deux ne domine l'autre sur tous les corpus, les fusionner par reciprocal rank fusion [17], une méthode simple qui combine les classements sans supposer que les scores sont comparables, donne presque toujours de meilleurs résultats que l'une ou l'autre seule.

Les embeddings eux-mêmes ont beaucoup évolué. L'approche dite de récupération contextuelle (enrichir chaque chunk d'un résumé de son contexte dans le document parent avant de le vectoriser) a montré des gains substantiels sur les corpus longs en 2024-2025, popularisée notamment par Anthropic et reprise dans plusieurs pipelines de production; nous pouvons également citer la méthode proposée par LightOn (NextPlaid) qui représente un passage par plusieurs vecteurs. Les modèles d'embedding modernes

(Voyage-3.5, Qwen3-Embedding, BGE-M3, Jina v4) sont entraînés sur des corpus multilingues et multifonctionnels bien plus larges que leurs prédécesseurs et dominent les benchmarks MTEB et MMTEB. Quant à l'interaction tardive (comparer les représentations token à token plutôt qu'un vecteur agrégé) elle reste pertinente pour les corpus où la granularité fine des correspondances compte : ColBERTv2 associé au moteur PLAID en est la version de production, avec des vitesses d'inférence comparables aux systèmes à vecteur unique. Le chunking, quant à lui, détermine ce qu'on récupère : découper les documents en fenêtres fixes est rapide mais corrompt les structures (tableaux, listes, raisonnements qui s'étendent sur plusieurs paragraphes) ; le chunking adaptatif, calé sur les frontières sémantiques et structurelles du document, coûte plus cher à l'indexation mais réduit les reconstructions absurdes.

Une fois les candidats rappelés, le reranking par cross-encodeur les retrie. Là où l'embedding compare des vecteurs pré-calculés (vite, mais sans contact direct entre la requête et chaque passage) le cross-encodeur relit la requête et le passage ensemble et note leur pertinence réelle. C'est plus lent, mais on ne le fait que sur les cinquante ou cent candidats déjà filtrés. L'interaction tardive de ColBERT [16] occupe un compromis intermédiaire : comparer les représentations token à token par similarité maximale (MaxSim), ce qui capture des correspondances que le vecteur agrégé rate, sans le coût du cross-encodeur complet. La compression contextuelle fait ensuite le ménage : plutôt que de passer tous les passages récupérés au modèle de génération, elle ne conserve que les fragments directement utiles au raisonnement, réduisant la fenêtre de contexte et limitant les interférences.

Enfin, côté sortie, des mécanismes de vérification et de calibration s'imposent progressivement comme des standards de production. Des modèles vérificateurs jugent si chaque affirmation générée est réellement ancrée dans les passages récupérés détectant les glissements où le modèle extrapole au-delà de ses sources. L'estimation d'incertitude et les scores de confiance apprennent au système à s'abstenir plutôt qu'à inventer : une réponse « je ne sais pas » coûte moins cher qu'une hallucination non détectée. La sélection d'outils, enfin (choisir entre index lexical, dense, graphe ou recherche web

selon la nature de la requête) s'apprend par renforcement ou par routage explicite (self-route), et c'est là que se concentre une large part de l'innovation de 2026.

Presque chacun de ces mécanismes descend en droite ligne d'un chapitre de manuel de recherche d'information. Ce qui a changé, c'est que la machine les orchestre seule et que quand l'un d'eux est absent ou mal réglé, l'échec est silencieux.

Ce que les interfaces savaient

Les facettes, les entités surlignées, les graphes n'ont jamais quitté les produits (l'e-commerce tourne aux facettes, les fiches d'entités habitent encore les pages de résultats) ils ont quitté le récit. La vague générative a effacé de la mémoire discursive les interfaces que la recherche d'information avait mis trente ans à concevoir : navigation à facettes pour découper un corpus par dimensions, NER pour surligner personnes, lieux et montants directement dans les documents, graphes d'entités pour les relations, frises chronologiques et vues d'ensemble pour orienter avant la plongée. Leur contrat commun (le système structure et oriente, l'humain lit et tranche, la provenance reste intacte) a été rompu par le paragraphe généré qui replace la lecture.

Le contrat à assembler tient en une ligne :générer pour orienter, citer pour prouver, laisser lire pour décider; et deux choses manquent encore.

- La première est comportementale : que vérifier coûte moins cher que croire ; les citations existent dans les meilleurs produits, mais la préférence révélée montre qu'on ne les suit pas (trafic référé effondré de 96 %).
- La seconde est l'assemblage lui-même : personne n'a encore shippé à grande échelle un système où synthèse fluide, entités surlignées, graphe navigable, facettes dynamiques et passages sources cohabitent dans une même vue.

Conclusion

Le retrieval suit le chemin de SQL: une commodité que plus personne ne vendra et sur laquelle tout reposera, derrière des interfaces standardisées comme MCP, pendant que la valeur migre vers l'orchestrateur qui raisonne, vérifie et budgète. La différence entre les grands modèles de 2026 ne se joue déjà plus sur la récupération mais sur la planification, le raisonnement multi-étapes, la sélection d'outils et la vérification. Les catégories technologiques meurent rarement d'échec ; elles meurent de succès, en devenant invisibles. Le RAG disparaîtra en tant que mot. La recherche d'information, elle, aura survécu à un buzzword de plus et si les cinq prochaines années sont bien jouées, elle en sortira avec les interfaces qu'elle méritait depuis le début.

Références

[1] Lewis, P., Perez, E., Piktus, A., et al. — *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*, NeurIPS 2020 (arXiv:2005.11401).

[2] Shah, C. & Bender, E. M. — *Situating Search*, CHIIR 2022.

[3] Magesh, V., et al. (Stanford RegLab/HAI) — *Hallucination-Free? Assessing the Reliability of Leading AI Legal Research Tools*, Journal of Empirical Legal Studies, 2025.

[4] Lindemann, N. — *Chatbots, search engines, and the sealing of knowledges*, AI & Society, 2024.

[5] Jiang, P., Rayan, A., Dow, S. P. & Xia, H. — *Graphologue: Exploring Large Language Model Responses with Interactive Diagrams*, UIST 2023 (arXiv:2305.11473).

[6] Suh, S., Min, B., Palani, S. & Xia, H. — *Sensecape: Enabling Multilevel Exploration and Sensemaking with Large Language Models*, UIST 2023 (arXiv:2305.11483).

[7] Eyuboglu, S., Arora, S., Ré, C., et al. — *Cartridges: Lightweight and General-Purpose Long Context Representations via Self-Study*, juin 2025 (arXiv:2506.06266).

[8] Subramanian, S., et al. (Amazon Science) — *Keyword Search is All You Need: Achieving RAG-Level Performance without Vector Databases using Agentic Tool Use*, AAAI 2026 (arXiv:2602.23368).

[9] Upadhyay, S., et al. — *Overview of the TREC 2025 RAG Track*, 2025 (arXiv:2603.09891).

[10] Georgiou, A. — *Spatially-Grounded Document Retrieval via Patch-to-Region Relevance Propagation*, 2025 (arXiv:2512.02660).

[11] Zou, W., Geng, R., Wang, B. & Jia, J. — *PoisonedRAG: Knowledge Corruption Attacks to Retrieval-Augmented Generation of Large Language Models*, USENIX Security 2025 (arXiv:2402.07867). Et Yu, X., et al. — *Cordon-MAS: Defending RAG-Based Multi-Agent Systems Against Confundo Poisoning Attacks*, 2026 (arXiv:2605.26754).

[12] Chhikara, P., et al. — *Mem0: Building Production-Ready AI Agents with Scalable Long-Term Memory*, arXiv:2504.19413, ECAI 2025.

[13] *From Web Search towards Agentic Deep Research: Incentivizing Search with Reasoning Agents*, 2025 (arXiv:2506.18959, douze institutions).

[14] Rocchio, J. J. — *Relevance Feedback in Information Retrieval*, in Salton (dir.), The SMART Retrieval System, Prentice-Hall, 1971.

[15] Bates, M. J. — *The Design of Browsing and Berrypicking Techniques for the Online Search Interface*, Online Review, 13(5), 1989.

[16] Santhanam, K., Khattab, O., Saad-Falcon, J., Potts, C. & Zaharia, M. — ColBERTv2: Effective and Efficient Retrieval via Lightweight Late Interaction, NAACL 2022 (arXiv:2112.01488). Et Santhanam, K., et al. — PLAID: An Efficient Engine for Late Interaction Retrieval, CIKM 2022 (arXiv:2205.09707).

[17] Cormack, G. V., Clarke, C. L. A. & Büttcher, S. — *Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods*, SIGIR 2009.

[18] Schwarcz, D., et al. — *AI-Powered Lawyering: AI Reasoning Models, Retrieval-Augmented Generation, and the Future of Legal Practice*, 2025 (prépublication SSRN).

[19] Hong, K., Troynikov, A. & Huber, J. — Context Rot: How Increasing Input Tokens Impacts LLM Performance, rapport technique, Chroma Research, juillet 2025. https://trychroma.com/research/context-rot

[20] Edge, D., Trinh, H., Cheng, N., Bradley, J., Chao, A., Mody, A., Truitt, S. & Larson, J. (Microsoft Research) — From Local to Global: A Graph RAG Approach to Query-Focused Summarization, arXiv:2404.16130, avril 2024.

[21] Respecting Temporal-Causal Consistency: Entity-Event Knowledge Graphs for Retrieval-Augmented Generation, arXiv:2506.05939 (et travaux connexes Wang et al., 2025 ; Chen et al., 2025).

[22] Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. Zep: A temporal knowledge graph architecture for agent memory. arXiv preprint arXiv:2501.13956, 2025.

Les opinions exprimées dans cet article sont strictement personnelles et ne reflètent pas nécessairement celles de mon employeur. Les contenus sont fournis à titre informatif et ne constituent pas un conseil juridique. Cet article explore des concepts architecturaux émergents et analyse des tendances de marché.

Eric Blaudez - AI Architect | Responsible AI · AI Act · AI Governance | Bringing R&D AI to Production (TRL 3→6) | EU & Sovereign Programs

Le RAG n'a (presque) rien inventé

Les opinions exprimées sur ce site sont strictement personnelles et ne reflètent pas nécessairement celles de mon employeur. Les contenus sont fournis à titre informatif et ne constituent pas un conseil juridique.