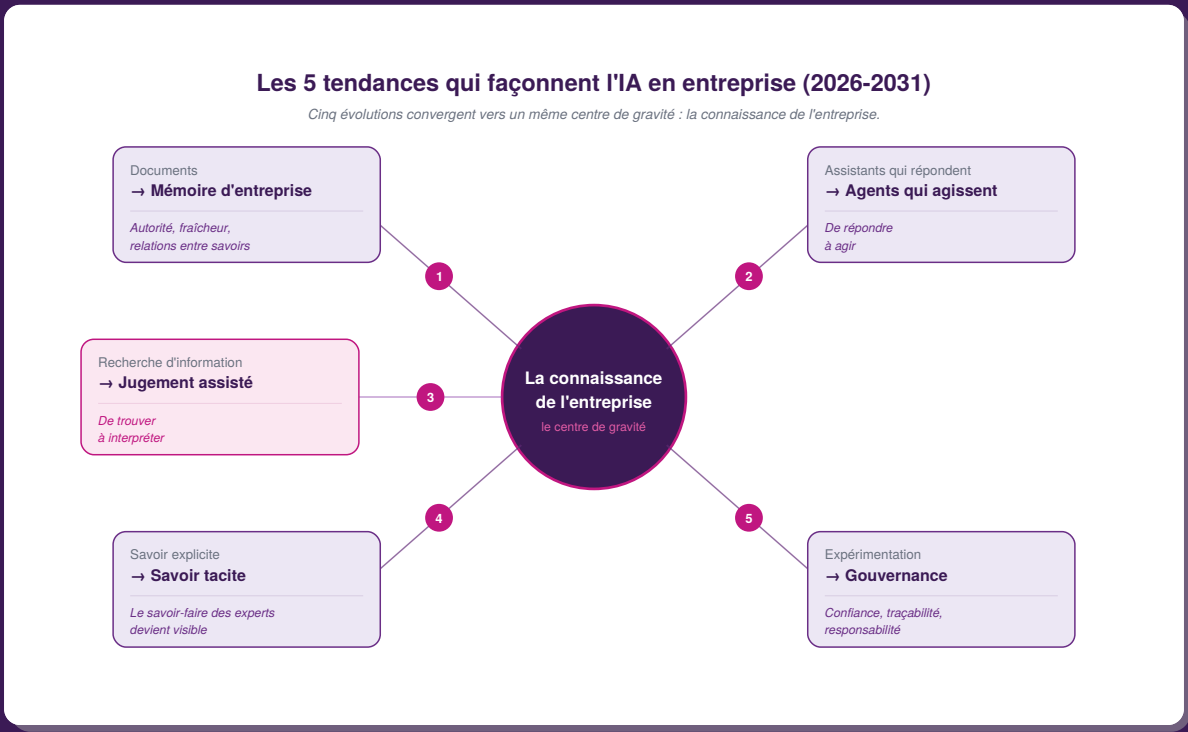
PROSPECTIVE • 2026 - 2031

Les cinq tendances qui façonnent l'IA en entreprise

Toutes convergent vers un même centre de gravité : la connaissance de l'entreprise.



Frédéric Ros • Technopole d'Orléans — LAB'IA Loire Valley

INTELLIGENCE ARTIFICIELLE • ENTREPRISE • GESTION DU SAVOIR

Pourquoi l’intelligence artificielle cale si souvent dans les entreprises

Le problème n’est pas l’IA, c’est le savoir de l’entreprise.

Pourquoi l’IA échoue moins par manque d’intelligence que par manque de connaissance.

Un parcours, du moteur de recherche à l’IA qui raisonne, pour comprendre ce qui se joue vraiment.

Frédéric Ros

Technopole d’Orléans — LAB’IA Loire Valley

Article de fond • Lecture ~25 min • June 9, 2026

© 2026 Frédéric Ros — Technopole d’Orléans, LAB’IA Loire Valley. Tous droits réservés. Toute reproduction, représentation ou diffusion, totale ou partielle, est interdite sans autorisation écrite préalable de l’auteur.

AVANT DE COMMENCER

L'intelligence artificielle est un territoire immense : apprentissage automatique, vision par ordinateur, robotique, modèles prédictifs, et bien d'autres continents. Cet article n'en explore qu'une presqu'île, mais c'est celle qui occupe aujourd'hui le plus d'entreprises : faire répondre une IA à partir de *leurs propres documents*. Nous laisserons donc volontairement de côté l'entraînement des modèles, les réseaux de neurones et la science qui les fait fonctionner – non qu'ils soient secondaires, mais parce que les comprendre n'est pas nécessaire pour saisir *pourquoi* tant de projets calent, ni *comment* les réussir. Aucun prérequis technique : si vous avez déjà utilisé Google et pesté contre un chatbot, vous avez tout ce qu'il faut pour lire ce qui suit.

L'IDÉE À RETENIR

On nous promet depuis des années que l'intelligence artificielle va transformer le travail. Dans beaucoup d'entreprises, pourtant, les projets s'enlisent. La raison n'est presque jamais celle qu'on croit : ce n'est pas que l'IA soit « trop bête ». C'est qu'on lui donne mal à manger. Cet article raconte pourquoi – et, chemin faisant, explique simplement comment fonctionnent les moteurs de recherche, les chatbots, les grands modèles d'IA, et cette technique au cœur de tout : le *RAG*.

Tout commence par une question

Vous tapez une question dans Google, et en une fraction de seconde vous recevez une liste de pages. C'est devenu si banal qu'on oublie l'exploit. Derrière ce geste se cachent deux opérations distinctes, et il faut les séparer parce que toute la suite en dépend. D'abord, Google **retrouve** : parmi des milliards de pages, il sélectionne celles qui parlent de votre sujet. Ensuite, il **classe** : il décide lesquelles afficher en premier. Retrouver, puis classer – comme un libraire qui sait d'abord dans quel rayon chercher, puis quel livre vous tendre en premier. La vraie trouvaille de Google, à la fin des années 1990, fut la seconde étape. Comment deviner qu'une page mérite plus de confiance qu'une autre ? L'idée géniale : une page importante est une page vers laquelle *beaucoup d'autres pages importantes pointent*. Comme une réputation : on fait davantage confiance à quelqu'un que des gens fiables recommandent. En suivant le réseau des liens, Google fait remonter le meilleur sans même « lire » le contenu.

LE MOT JUSTE

Cette méthode porte un nom resté célèbre : **PageRank**. Retenez l'intuition : la qualité se mesure aux *relations* entre les pages, pas seulement à leur contenu.

Mais remarquez ce que Google *ne* fait *pas* : il ne vous donne pas la réponse. Il vous donne des *liens*. C'est vous qui ouvrez trois onglets, lisez, recoupez, et fabriquez la réponse dans votre tête. Google retrouve et classe ; la *synthèse*, c'est encore l'humain.

Le chatbot d'avant : une machine à règles

Prenons maintenant un objet que tout le monde a détesté au moins une fois : le chatbot d'ancienne génération. Le serveur vocal qui vous lance « tapez 1 pour le service client », ou cette fenêtre de « dialogue » sur un site qui répond toujours à côté. Comprendre comment il marche éclaire tout le reste. Ces systèmes ne comprennent rien à ce que vous dites. Ils fonctionnent par **règles écrites à la main**. Quelque part, un humain a prévu : *si* le message contient le mot « facture », *alors* répondre ceci ; *si* il contient « remboursement », *alors* cela. C'est un gigantesque arbre de *si… alors…*, parfois des milliers de branches.

> **LE MOT JUSTE**
>
> On parle de systèmes **à base de règles** (ou *à base de mots-clés*). La machine ne saisit pas le *sens* : elle repère des mots et suit un scénario écrit d'avance.

D'où leur fragilité, que vous avez forcément vécue. Vous écrivez « je n'ai pas reçu mon argent » ; le mot « remboursement » n'y est pas, donc la machine ne reconnaît rien et vous renvoie le menu de départ. Elle ne sait répondre qu'aux questions *exactement prévues*, formulées *exactement* comme prévu. Hors du scénario, elle est aveugle.

> **L'IDÉE À RETENIR**
>
> Le chatbot d'ancienne génération ne comprend pas le langage : il l'imite par des règles. Sa limite n'est pas un défaut de réglage, c'est sa nature même. Pour franchir cette limite, il fallait une machine capable de saisir le *sens*, pas seulement les *mots*.

Entre une machine qui comprend vraiment

Cette machine est arrivée. Ce sont les modèles comme ChatGPT, Claude ou Gemini. On les appelle des *grands modèles de langage*, et ils ont rendu possibles les chatbots de nouvelle génération – ceux à qui vous pouvez parler normalement, qui saisissent « je n'ai pas reçu mon argent » sans qu'on ait écrit la règle correspondante. Pour comprendre ce qu'ils sont, une image vaut mieux qu'une définition. Imaginez un étudiant exceptionnel ayant lu une quantité vertigineuse de textes : livres, articles, conversations. À force, il a développé un sens très sûr de la façon dont les idées et les mots s'enchaînent. Posez-lui une question, il répond dans une langue fluide, construite, convaincante.

> **LE MOT JUSTE**
>
> On dit que ces modèles sont entraînés à **prédire le mot suivant**. Cela paraît modeste, mais en s'exerçant sur des montagnes de textes, la machine finit par manier le raisonnement, le style, les nuances. Elle ne récite pas : elle *compose*.

La différence avec le chatbot d'avant est abyssale. L'ancien suivait un arbre de règles figé. Le nouveau *généralise* : il n'a pas besoin qu'on ait prévu votre formulation, il en saisit l'intention. Et là où Google vous tendait des documents à lire, le modèle, lui, vous *parle* : il ne se contente pas de retrouver, il **rédige**.

SOUS LE CAPOT

« Comprend »-il vraiment ? Dire que le modèle « comprend » ou « saisit le sens » est une simplification utile, et nous l'emploierons dans tout cet article. Mais soyons exacts : les chercheurs débattent encore de la nature de cette « compréhension ». Ce qui est certain, c'est que la machine manipule le langage assez finement pour produire des comportements qui *ressemblent* à de la compréhension – ce qui suffit largement à notre propos. Gardez simplement en tête que derrière le mot commode se cache une question encore ouverte.

C'est prodigieux. Mais c'est là que les ennuis commencent.

Figure 1. L'étudiant brillant : fluide, mais pas toujours factuel. Sa connaissance s'arrête à sa date d'entraînement, et il peut inventer des réponses plausibles.

Le talon d'Achille : il ne sait que ce qu'il a lu

Notre étudiant brillant a un défaut majeur, et il est double. D'abord, sa connaissance s'arrête au jour où il a fini ses lectures. Tout ce qui a suivi lui est inconnu : ni les résultats de votre entreprise ce trimestre, ni la procédure modifiée la semaine dernière, ni le contrat signé hier. Ensuite – plus sournois – quand il ignore, il ne le dit pas toujours. Comme il a appris à produire des réponses *plausibles*, il peut inventer un détail, un chiffre, une date, avec le plus parfait aplomb. Il ne ment pas au sens humain : il comble un blanc par ce qui *ressemble* à une bonne réponse.

LE MOT JUSTE

Ce phénomène a un nom : l'**hallucination**. Le modèle énonce avec assurance une chose fausse. Ce n'est pas un bug rare : c'est la conséquence directe de sa nature – il a été conçu pour être fluide, pas pour être exact.

Pour un usage personnel, on s'en accommode. Mais une entreprise ne peut pas bâtir un service client, un assistant juridique ou un outil de décision sur une machine qui invente et qui

ignore tout de ses données récentes. Il fallait ancrer ces réponses dans des faits réels, à jour et vérifiables.

La solution : ne pas tout lui faire apprendre, mais lui donner à lire

Première idée, naturelle : puisque le modèle ne connaît pas nos documents, ré-entraînons-le sur les nôtres. Faisons-lui « avaler » nos milliers de fiches internes. Tentant, mais cela revient à demander à notre étudiant de mémoriser par cœur tout le classeur de l'entreprise. La mémoire est diffuse : il confondra deux procédures, restituera un chiffre de travers, et si une fiche change demain, il faudra tout lui faire réapprendre. Sans compter qu'il sera incapable de dire *dans quel document* il a trouvé sa réponse. Il existe une approche bien plus maligne. Plutôt que de gaver l'étudiant, on lui laisse son intelligence intacte et on pose à côté de lui **un classeur bien rangé**. À chaque question, il ouvre le classeur à la bonne page, lit ce qui s'y trouve, et formule sa réponse *à partir de ce qu'il vient de lire* – en citant la page.

L'IDÉE À RETENIR

On ne demande pas à la machine de *tout savoir*. On lui demande de savoir *lire le bon document au bon moment* : aller chercher les documents pertinents, puis les lui donner à lire juste avant qu'elle réponde.

LE MOT JUSTE

Cette technique s'appelle le **RAG**, pour *Retrieval-Augmented Generation* – « génération augmentée par la récupération ». En clair : on *récupère* les bons documents, puis on *augmente* la réponse du modèle avec leur contenu.

Les avantages tombent d'eux-mêmes. Une procédure change ? On remplace une feuille dans le classeur, le modèle ne réapprend rien. Une réponse douteuse ? Elle cite sa source, on vérifie. Et comme le modèle s'appuie sur un texte réel sous les yeux plutôt que sur ses souvenirs, il hallucine beaucoup moins. Vous l'avez sans doute déjà vu à l'œuvre : quand un assistant « va chercher sur internet » et cite ses sources, il fait du RAG – son classeur, ce jour-là, c'est le web. La nouveauté, en entreprise, c'est qu'on remplace ce classeur public par *vos* documents privés.

SOUS LE CAPOT

Pourquoi maintenant, et pas en 2015 ? On pourrait croire le RAG né d'une percée soudaine. C'est l'inverse : presque toutes ses pièces existaient déjà depuis des années. Les moteurs de recherche savaient retrouver et classer ; les empreintes de sens (les vecteurs) étaient connues ; les bases pour les ranger aussi. Il manquait une seule pièce, mais décisive : un *rédacteur* capable de lire ces documents et d'en synthétiser une réponse en langage naturel. C'est l'arrivée des grands modèles, vers 2022-2023, qui a fourni cette pièce manquante. Le RAG n'est donc pas une invention isolée : c'est la **rencontre** entre une recherche documentaire déjà mûre et un rédacteur enfin capable. D'où sa diffusion fulgurante – le terrain était prêt depuis longtemps.

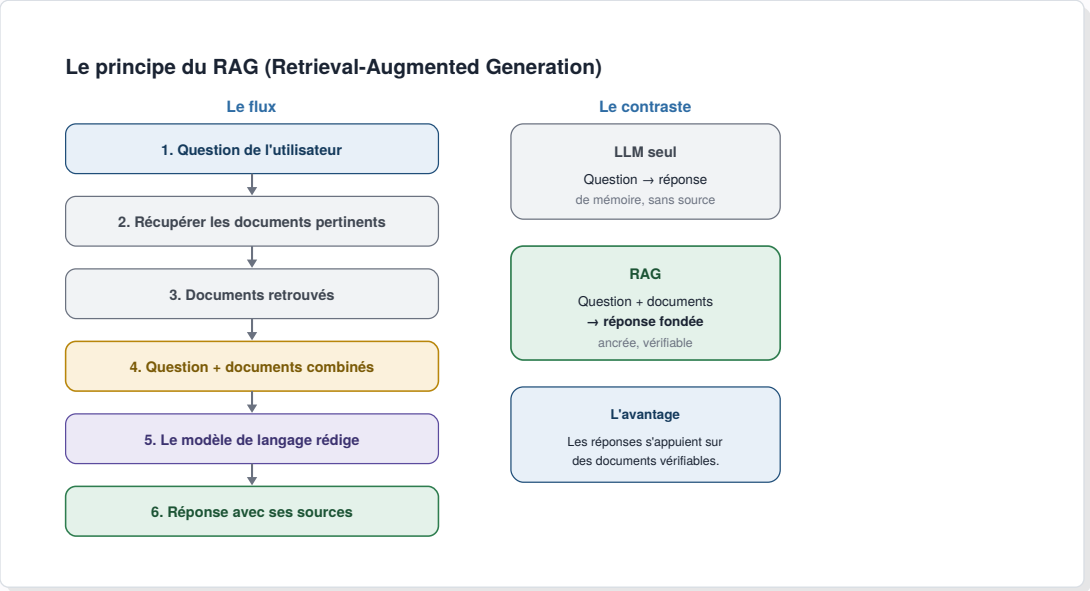


Figure 2. Le principe du RAG. À gauche, le flux complet : la question déclenche une récupération de documents, qui sont donnés au modèle pour produire une réponse *sourcée*. À droite, le contraste avec le LLM seul : le RAG ancre la réponse dans des documents vérifiables.

Sous le capot : que se passe-t-il vraiment ?

Montons d'un cran. « Ranger un classeur » et « chercher la bonne page », concrètement, comment une machine fait-elle ? Le processus se déroule en deux temps. Un premier temps, *une fois pour toutes*, prépare le classeur. Un second temps, *à chaque question*, s'en sert. **Préparer le classeur (en coulisses, à l'avance).**

1. **Découper.** On ne range pas des documents de cent pages d'un bloc : on les coupe en morceaux digestes, de la taille d'un paragraphe. Chaque morceau doit rester compréhensible tout seul.
2. **Transformer en « empreintes ».** Voici le cœur technique. À chaque morceau de texte, la machine associe une suite de nombres qui résume son *sens*. Deux textes qui parlent de la même chose – même avec des mots différents – reçoivent des empreintes *proches*.
3. **Ranger.** Toutes ces empreintes sont stockées dans une base spéciale, conçue pour retrouver très vite les plus ressemblantes.

> **LE MOT JUSTE**
>
> Cette « empreinte de sens » s'appelle un **vecteur** (ou *embedding*). L'idée clé : on transforme le *sens* d'un texte en *position dans l'espace*. Des textes au sens voisin se retrouvent voisins. La base qui les stocke est une **base vectorielle**.

Répondre à une question (à chaque fois).

1. **Transformer la question** en empreinte, de la même façon.

2. **Retrouver** dans la base les quelques morceaux dont l'empreinte est la plus proche de celle de la question. Ce sont les passages les plus susceptibles de contenir la réponse.
3. **Donner à lire** ces passages au modèle, accompagnés de la question.
4. **Rédiger** : le modèle compose sa réponse à partir de ces passages, et peut citer d'où ils viennent.

SOUS LE CAPOT

Pourquoi ça marche : le sens devient géométrie. Pourquoi des *nombres* pour représenter du sens ? Parce qu'une machine ne compare pas des idées, elle calcule des distances. L'astuce des empreintes est de placer chaque texte dans un vaste espace de telle sorte que la *proximité* y traduise la *parenté de sens* : « automobile » atterrit près de « voiture », loin de « banane ». Dès lors, « chercher un document proche de la question » devient « trouver le point le plus proche » – un calcul que l'ordinateur fait en un éclair, même sur des millions de documents. Le sens, transformé en position, devient calculable. C'est cette idée qui permet la **recherche par le sens**, là où Google, à l'origine, cherchait surtout par les mots.

Le lien avec Google

Vous l'avez peut-être deviné. Pour donner à lire le bon document au modèle, il faut d'abord le *retrouver* et le *classer*. Exactement les deux talents de Google.

	Google	Le RAG
Retrouver	les pages du web	les bons documents internes
Classer	le meilleur en premier	le plus pertinent en premier
Répondre	*c'est vous qui lisez*	**le modèle rédige pour vous**

Un système RAG, c'est donc un moteur de recherche *branché sur* un rédacteur. Google s'arrêtait à la deuxième ligne ; le RAG ajoute la troisième. Et c'est là que se cache le malentendu qui fait échouer tant de projets.

Pourquoi 80 % des projets IA ne passent pas le pilote

Posons la question que se pose tout dirigeant ayant signé un chèque : *pourquoi mon projet d'IA à plusieurs centaines de milliers d'euros n'a-t-il rien donné ?* La démonstration était bluffante ; six mois plus tard, l'assistant est abandonné. Ce scénario est si fréquent qu'il est presque la règle. Et la cause, le plus souvent, n'a rien à voir avec l'IA elle-même.

L'IDÉE À RETENIR

La vérité qui dérange : un projet RAG échoue rarement à cause du modèle ou de la technique. Il échoue à cause de l'**état du savoir de l'entreprise**. On a investi dans un moteur sophistiqué pour explorer un classeur en désordre.

Quatre obstacles, presque toujours les mêmes, expliquent l'enlisement, et aucun ne se règle en changeant de modèle :

- **Des documents dispersés.** Le savoir est éclaté entre des serveurs partagés, des boîtes mail, des messageries, des têtes. Le classeur n'existe pas vraiment : il faut d'abord le constituer.
- **Une qualité documentaire faible.** Procédures qui se contredisent, PDF de 2017 jamais mis à jour, cinq versions du même document sans qu'on sache laquelle fait foi. Le RAG va puiser là-dedans en toute confiance.
- **Des connaissances qui ne sont écrites nulle part.** Les règles réellement appliquées vivent dans la tête des experts, pas dans les fiches.
- **Une absence de gouvernance.** Personne n'est responsable de tenir le classeur à jour, d'arbitrer les contradictions, de retirer l'obsolète.

L'IDÉE À RETENIR

À retenir, et à répéter à qui veut lancer un projet : **un RAG n'est pas un remède à une mauvaise gestion documentaire.** Il accélère l'accès à la connaissance *existante* ; il ne crée pas la connaissance *manquante*. Brancher une IA sur un savoir en désordre, c'est automatiser le désordre, plus vite, et avec plus d'aplomb.

Figure 3. Les quatre causes récurrentes d'échec convergent vers une même vérité : l'IA ne crée pas la connaissance manquante.

Le plus profond de ces obstacles mérite qu'on s'y arrête, car c'est sur lui que portera la fin de cet article. Prenez une situation que tout praticien reconnaît : un document dit **A**, un autre

dit **B**, et pourtant tout le monde, dans le service, sait qu'il faut appliquer **C**. Pourquoi C ? Parce qu'un expert le sait – par expérience, par usage, par mémoire des cas passés. Ni le document, ni le RAG, ni le modèle ne peuvent le deviner : l'information n'est écrite *nulle part*. Gardez ce cas en tête ; nous y reviendrons.

> **SOUS LE CAPOT**
>
> **Le coût, ce grand absent des démonstrations.** Une maquette ne coûte rien ; un système en production, si. Transformer les documents en empreintes (les *embeddings*) a un coût, payé à chaque mise à jour du corpus. Chaque question posée au modèle a un coût. Et un agent qui réfléchit en cinq étapes coûte environ *cinq fois* une requête simple, puisque chaque boucle est un nouvel appel facturé. À cela s'ajoute la maintenance : tenir le classeur à jour est un travail permanent, pas un projet qu'on livre et qu'on oublie. Beaucoup de projets calent là – non parce qu'ils ne marchent pas, mais parce qu'ils coûtent plus qu'on ne l'avait prévu.

L'autre échec, plus technique : le RAG naïf

Supposons maintenant le classeur en bon état. Une seconde famille d'échecs guette, purement technique celle-là. Quand une équipe construit un assistant interne, toute l'attention se porte sur le modèle – le « cerveau ». On choisit le plus puissant, on soigne les questions qu'on lui pose. Et on néglige la partie ingrate : la façon de ranger et de chercher dans le classeur.

> **L'IDÉE À RETENIR**
>
> Le maillon faible d'un RAG n'est presque jamais le modèle. C'est l'étape silencieuse qui va chercher les documents. Un cerveau génial à qui l'on tend la mauvaise page rédigera une réponse magnifique... et fausse.

C'est le piège du **RAG naïf** : une version minimale où l'on découpe grossièrement, on range, et on espère. Cela marche à merveille sur une démonstration de dix pages. Puis on branche les dix mille documents réels de l'entreprise, et la qualité s'effondre. Quelques causes, très concrètes :

- **On découpe mal.** Couper au milieu d'une phrase ou d'un tableau, c'est ranger des demi-pages incompréhensibles. Le modèle reçoit un fragment privé de son contexte.
- **On range mal les documents complexes.** Un PDF avec tableaux, schémas, colonnes : aplati bêtement en texte, l'information clé se perd avant même d'entrer dans le classeur.
- **On cherche sur les mots, pas sur le sens.** L'utilisateur demande « congés », le bon document parle d'« absences » : un système mal conçu passe à côté.
- **On donne tout, sans trier finement.** On tend dix documents *à peu près* pertinents au lieu des deux *vraiment* utiles. Noyé, le modèle s'accroche au mauvais.

Aucun de ces problèmes ne se voit sur la maquette ; tous explosent en conditions réelles. Et comme l'attention était braquée sur le cerveau, l'équipe conclut souvent, à tort, que « l'IA n'est pas prête ». Alors que c'est le classeur, et la façon d'y chercher, qu'on n'avait pas pensé.

L'IDÉE À RETENIR

Un exemple qui résume tout. Un employé demande : « combien de jours de congé pour la naissance de mon enfant ? » La bonne réponse est dans une note RH, mais sous l'intitulé « congé de paternité et d'accueil », dans un tableau, au milieu d'un PDF de quarante pages. Le RAG naïf échoue trois fois d'affilée : il a coupé le tableau en miettes (mauvais découpage), il cherche le mot « naissance » qui n'apparaît pas (recherche par les mots), et il finit par tendre au modèle un paragraphe sur les congés *sans solde*, à peu près ressemblant. Le modèle, docile, rédige une réponse impeccable... et fausse. Chacune des parades ci-dessous répare l'un de ces trois ratés.

Et ces ratés se paient cher, d'une façon qu'on sous-estime presque toujours : en **confiance**. Un assistant interne ne se juge pas sur sa réponse moyenne, mais sur sa pire. Un seul mauvais conseil RH, une seule réponse juridique erronée, et l'utilisateur cesse d'y croire – définitivement. Il retournera demander à son collègue, comme avant. La confiance se gagne lentement, sur des centaines de bonnes réponses, et se perd en une seule mauvaise.

L'IDÉE À RETENIR

La fiabilité d'un assistant n'est pas une moyenne, c'est un seuil. En dessous d'un certain niveau de confiance, les utilisateurs l'abandonnent – et un outil que personne n'utilise est un échec, quelle que soit sa prouesse technique. C'est pourquoi mieux vaut un assistant qui *sait dire « je ne sais pas »* qu'un assistant brillant mais parfois faux.

Mieux faire : le sens plutôt que les mots, et les autres leviers

La bonne nouvelle : chacun de ces échecs a une parade connue. Passer du RAG naïf au RAG professionnel, ce n'est pas changer de cerveau – c'est un travail d'artisan sur le classeur. Voici les principaux leviers, dans l'ordre où ils répondent aux échecs ci-dessus. **Chercher par le sens.** On l'a vu sous le capot : grâce aux empreintes (les vecteurs), la machine rapproche « congés » et « absences » parce que leur *sens* est voisin, même sans mot commun. C'est le saut décisif par rapport à la recherche par mots-clés des vieux systèmes. **Mieux découper.** Plutôt que couper tous les 500 caractères, on respecte la structure du document : un paragraphe, une section, une idée par morceau. On parle de découpage *intelligent* – et c'est souvent ce qui sépare un RAG médiocre d'un bon. **Mieux lire les documents complexes.** Pour les tableaux, schémas et PDF mis en page, on utilise désormais des outils qui « regardent » la page comme le ferait un humain, au lieu de la réduire à du texte brut. **Reformuler la question.** La question de l'utilisateur est rarement la meilleure pour chercher. On peut la reformuler, la découper en sous-questions, ou en générer plusieurs versions, pour ratisser plus juste. **Trier avant de répondre.** Une fois une dizaine de passages retrouvés, une seconde étape les *re-classe* pour ne garder que les plus pertinents. On donne ainsi au modèle deux excellents passages plutôt que dix médiocres. **Mélanger résumés et texte.** Voici une idée plus subtile. Pour retrouver le bon morceau, on a parfois intérêt à chercher non pas dans le texte brut, mais dans un *résumé* de ce texte – plus court, plus net, donc plus facile à faire correspondre à une question. On range alors côte à côte plusieurs « vues » du même passage : le texte intégral pour la lecture finale, mais aussi un résumé, ou les questions auxquelles ce passage répond, pour la recherche. On

cherche sur la vue la plus pratique, et on lit la vue la plus complète. **Donner du contexte à chaque morceau.** Un paragraphe sorti de son chapitre perd son sens (« dans ce cas, le délai est de 30 jours » – mais quel cas ?). Une parade récente consiste à recoller à chaque morceau une phrase de contexte « ce passage provient de la procédure de remboursement, section retards », avant de le ranger. Le morceau redevient compréhensible seul.

LE MOT JUSTE

Cette dernière étape de tri fin s'appelle le **re-ranking**. Ranger plusieurs vues d'un même passage s'appelle la *multi-représentation*. Avec la recherche par le sens (*recherche sémantique*) et le découpage intelligent, ce sont les leviers qui transforment le plus souvent un prototype fragile en système fiable.

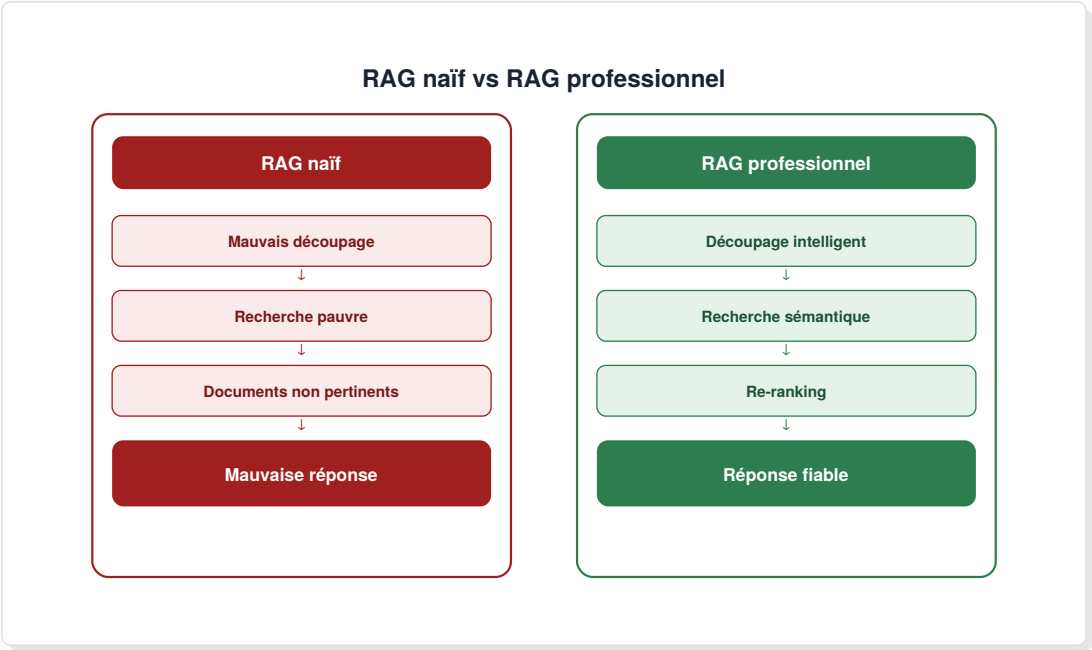


Figure 4. La même question, deux chaînes de traitement. À gauche, le RAG naïf accumule les ratés jusqu'à la mauvaise réponse ; à droite, le RAG professionnel répare chaque maillon pour aboutir à une réponse fiable.

Combiner les deux mondes, et relier les idées entre elles. Deux pistes plus avancées méritent d'être citées. D'une part, on peut *marier* la recherche par mots-clés (imbattable sur un numéro de référence exact) et la recherche par le sens (imbattable sur une reformulation) : c'est la *recherche hybride*. D'autre part, plutôt que de traiter chaque document isolément, on peut construire une carte des *relations* entre les notions qu'ils contiennent – qui est lié à quoi, quelle procédure découle de quelle norme. Le système ne cherche plus seulement des passages, il suit des fils entre les idées.

LE MOT JUSTE

Cette « carte des relations » entre les notions d'un corpus s'appelle un **graphe de connaissances**. C'est l'une des pistes les plus actives aujourd'hui pour des questions où la réponse n'est pas dans un seul document, mais dans la *mise en relation* de plusieurs.

SOUS LE CAPOT

Pourquoi le « GraphRAG » suscite tant d'intérêt. Imaginez la question : « quels projets sont touchés si le fournisseur X fait faillite ? » Aucun document ne contient la réponse. Elle se reconstitue en reliant : X fournit le composant Y, Y entre dans les produits Z1 et Z2, qui équipent les projets P1 et P3. La recherche classique, qui retrouve des passages *isolés* ressemblant à la question, est ici impuissante : il faut *suivre une chaîne*. En représentant le savoir comme un réseau de liens (le graphe), le système peut parcourir ces connexions de proche en proche. C'est ce saut – de « retrouver des textes » à « naviguer dans un réseau d'idées » – qui explique l'engouement.

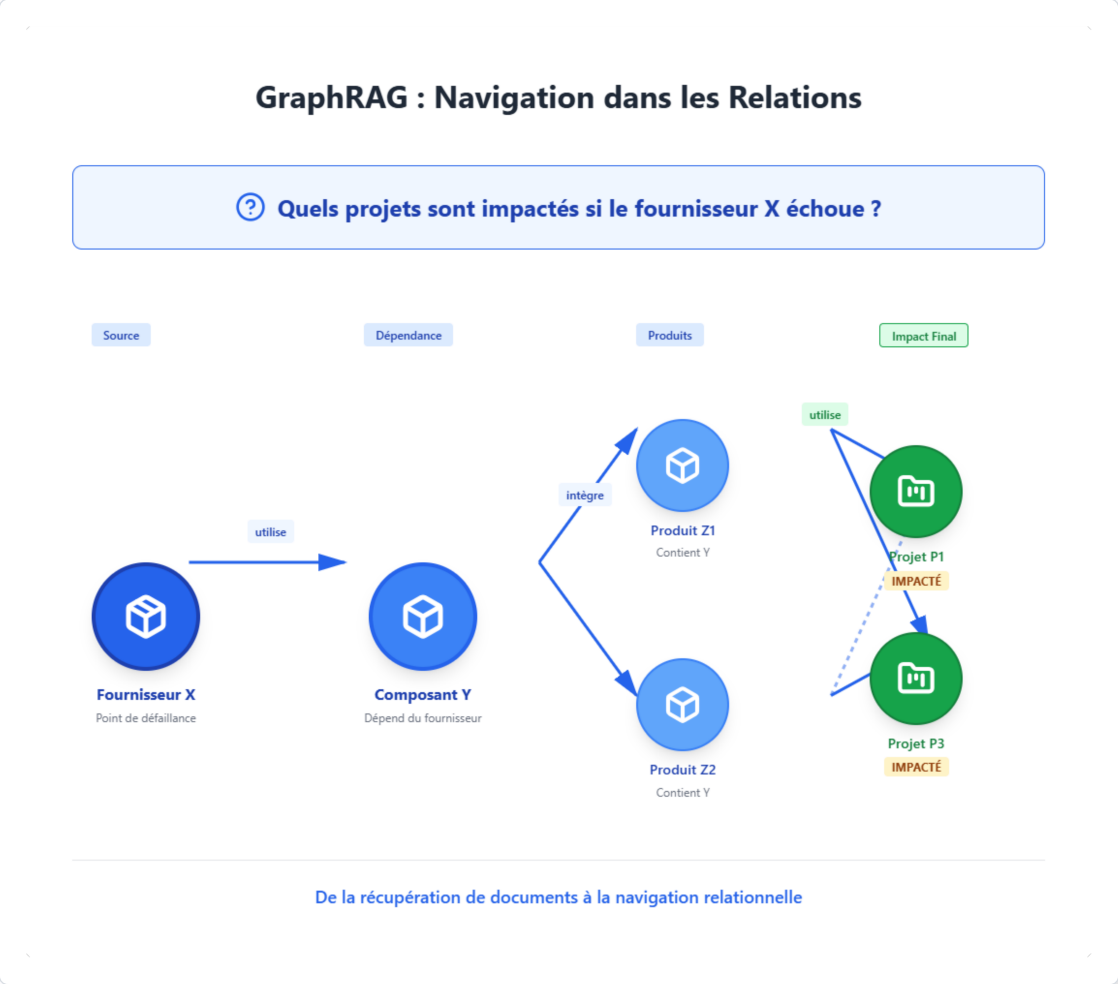


Figure 5. GraphRAG en action : aucune page ne contient la réponse, mais en suivant les liens (le fournisseur X alimente le composant Y, présent dans les produits Z1 et Z2, qui équipent les projets P1 et P3), le système reconstitue l'impact. De la récupération de documents à la navigation relationnelle.

Le cran au-dessus : les agents

Jusqu'ici, notre système suit toujours le même trajet : une question, une recherche, une réponse. En ligne droite. Mais certaines questions ne se laissent pas résoudre en une fois – « compare

les conditions de nos trois contrats fournisseurs et dis-moi lequel est le plus risqué » demande de chercher plusieurs fois, de comparer, de raisonner par étapes. Pour saisir la nouveauté, reprenons notre mouvement favori : la même bascule que pour les chatbots, mais appliquée à *l'action*. **L'agent d'avant : un automate qui suit un script.** Depuis longtemps, des programmes enchaînent des actions tout seuls : *si* telle condition, *alors* faire ceci, puis cela. Un thermostat, une chaîne de traitement automatique. Mais, comme le vieux chatbot, cet « agent » ne fait qu'exécuter un plan écrit d'avance par un humain. Hors de son script, il est perdu. **L'agent d'aujourd'hui : un modèle qui décide lui-même.** On confie désormais le volant au modèle de langage. On ne lui donne plus un script, mais un *objectif* et une *boîte à outils* – par exemple : « tu peux chercher dans le classeur, autant de fois que nécessaire ». Et c'est lui qui décide : je cherche d'abord le contrat A, puis le B, je compare, il me manque un détail, je relance une recherche, et enfin je conclus.

LE MOT JUSTE

Quand le modèle planifie, agit, observe le résultat et recommence en boucle jusqu'à atteindre son but, on parle de **RAG agentique** (ou d'*agent*). La recherche n'est plus une étape unique en ligne droite : elle devient un *outil* que le modèle actionne plusieurs fois, à sa guise.

SOUS LE CAPOT

La différence tient en un mot : *l'initiative*. Le RAG classique cherche une fois, puis répond. L'agent décide *combien* de fois chercher, *quoi* chercher ensuite en fonction de ce qu'il vient de lire, et *quand* il en sait assez pour s'arrêter. Cette liberté le rend puissant sur les questions complexes – mais aussi plus lent et plus coûteux, car chaque boucle est un appel supplémentaire. Tout l'art consiste à lui laisser juste ce qu'il faut d'initiative.

On boucle ainsi la boucle de cet article. La même rupture s'est jouée deux fois : des *règles écrites par l'humain* vers une *machine qui comprend* (les chatbots), puis d'un *script écrit par l'humain* vers une *machine qui décide* (les agents). À chaque fois, on est passé de « la machine fait exactement ce qu'on a prévu » à « la machine s'adapte à ce qu'elle rencontre ».

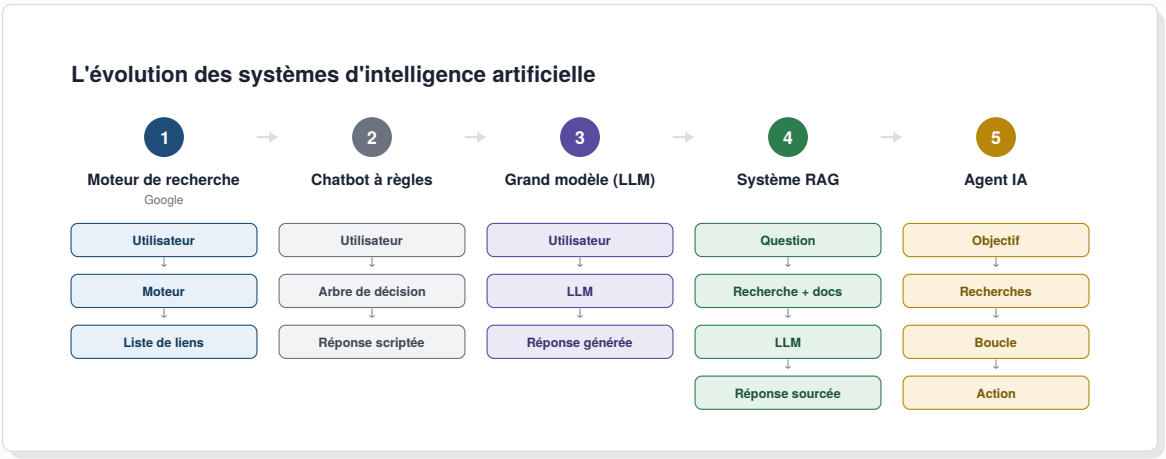


Figure 6. Cinq étapes d'une même histoire : du moteur de recherche au chatbot à règles, puis au modèle de langage, au RAG, et enfin à l'agent. À chaque étape, la machine gagne en autonomie – de la liste de liens à l'action.

Le vrai gain : automatiser des tâches, pas seulement répondre

Jusqu'ici, on a parlé d'un système qui *répond*. Mais le bénéfice le plus concret pour une entreprise n'est pas d'obtenir une belle réponse à l'écran : c'est de *faire faire* un travail. Répondre, c'est utile ; **agir**, c'est ce qui fait gagner du temps. Et c'est là que les agents, nourris par le RAG, changent vraiment la donne. Pour qu'un agent automatise une tâche, il lui faut deux choses : savoir *quoi faire* – la procédure, la règle, le précédent – et pouvoir *agir* – remplir un formulaire, router une demande, rédiger un courrier. Le RAG fournit la première : il va chercher dans le classeur de l'entreprise la connaissance juste et à jour, et la donne à l'agent pour qu'il agisse correctement. Sans RAG, l'agent agirait à l'aveugle, sur des règles génériques ou périmées. Avec RAG, il agit sur *vos* règles, telles qu'elles sont aujourd'hui.

> **L'IDÉE À RETENIR**
>
> Le RAG n'est pas une fin en soi : c'est le **carburant** de l'automatisation. Il apporte à l'agent la connaissance métier sans laquelle « agir » ne voudrait rien dire. L'agent est la main ; le RAG est la mémoire qui guide la main.

Rendons cela concret. Voici des tâches que ce duo automatise déjà bien, et le rôle du RAG dans chacune :

- **Traiter une demande de congé.** L'agent lit la demande, va chercher *via le RAG* la règle applicable (solde, ancienneté, période de fermeture), vérifie, puis valide ou explique le refus. Une tâche faite cent fois par jour, désormais instantanée.
- **Répondre à une question de premier niveau au support client.** « Comment réinitialiser mon mot de passe ? » : l'agent retrouve la procédure exacte dans la base interne et rédige la réponse, sources à l'appui. Les humains se concentrent sur les cas vraiment difficiles.
- **Vérifier la conformité d'un document.** Avant signature, l'agent confronte un contrat aux clauses-types et à la politique maison (retrouvées par le RAG) et signale ce qui manque ou cloche.
- **Préparer un dossier.** Pour une réclamation, l'agent rassemble automatiquement l'historique du client, la procédure applicable et les courriers précédents – le travail de collecte qui prenait vingt minutes à un humain.
- **Trier et router le courrier entrant.** Lire un message, comprendre de quoi il s'agit, l'envoyer au bon service avec un résumé : un tri qui s'appuie sur la connaissance de l'organisation pour ne pas se tromper de porte.

Dans tous ces cas, le gain de productivité est réel et *mesurable* : un temps de traitement qui s'effondre, des humains libérés des tâches répétitives pour les cas à valeur ajoutée. Mais – et c'est le point essentiel – remarquez ce que tous ces exemples ont en commun.

> **L'IDÉE À RETENIR**
>
> Ça marche quand la tâche est **cadrée et déterministe** : des règles claires, des étapes prévisibles, un résultat qu'on peut *vérifier*. Une demande de congé a une bonne réponse ; un contrat est conforme ou non. C'est cette possibilité de contrôle qui fait décoller la productivité – on délègue

sans crainte parce qu'on peut vérifier.

À l'inverse, dès que la tâche devient floue, ouverte, à fort enjeu de jugement, l'automatisation déraille. « Faut-il licencier ce salarié ? », « cette stratégie est-elle la bonne ? » : il n'y a pas de procédure à retrouver, pas de résultat vérifiable mécaniquement. Non que l'IA soit « trop bête » – mais parce qu'il n'existe plus de cadre garantissant qu'elle a raison.

SOUS LE CAPOT

La bonne ligne de partage n'est pas « tâche simple / tâche complexe », mais « résultat vérifiable / résultat incertain ». On **automatise** ce qu'on peut contrôler ; on se contente d'**assister** l'humain là où le jugement prime. Un bon déploiement commence presque toujours par les tâches cadrées et répétitives – celles où le retour sur investissement est immédiat et le risque faible – avant de s'aventurer vers les zones plus grises.

Ce critère – *pouvoir vérifier* – a une conséquence directe, qui nous mène à la dernière idée de cet article. Pour cadrer une tâche, encore faut-il savoir quelle procédure fait foi, quelle source est à jour, quel document fait autorité. Or cette connaissance-là, on va le voir, n'est écrite nulle part dans les documents.

La vraie frontière : la connaissance n'est pas dans les documents

Souvenez-vous du cas laissé en suspens : le document dit A, un autre dit B, et tout le monde applique C. C'est ici qu'il trouve sa réponse – et c'est, à notre sens, l'idée la plus importante de tout cet article. Tout ce qui précède partage une supposition que l'on formule rarement : *si les documents sont bien découpés, bien rangés et bien cherchés, le système trouvera la bonne information*. Toute l'énergie va donc à perfectionner la recherche. Mais une piste de réflexion émergente – et c'est l'hypothèse que nous explorons – suggère que cette supposition est incomplète, et que c'est précisément ce qui explique tant d'échecs. Observez un expert au travail : un juriste, un ingénieur qualité, un responsable RH face à ses dossiers. Il ne cherche pas seulement le document qui *ressemble* le plus à la question. Il sait, en plus, des choses que le texte ne dit pas lui-même :

- quels documents font **autorité**, et lesquels ne sont qu'indicatifs ;
- quelle procédure **prévaut** sur une autre en cas de conflit ;
- quelles sources sont **fiables**, et lesquelles sont **obsolètes** ;
- quels passages sont **critiques**, et lesquels ne sont qu'un décor.

L'IDÉE À RETENIR

Cette connaissance-là n'est écrite *nulle part dans les documents*. Elle vit dans la tête des experts : c'est leur *façon d'utiliser* le corpus. Or un système qui ne juge que la ressemblance entre la question et le texte en est, par construction, aveugle. Il retrouve souvent une information correcte – mais pas forcément celle que l'expert aurait choisie.

C'est peut-être l'angle mort des systèmes actuels. On a beaucoup appris à *chercher* dans les documents ; on a peu réfléchi à *représenter le savoir* qu'ils portent. Deux organisations peuvent posséder exactement les mêmes documents et les utiliser de façons radicalement différentes – parce que leur *usage* diffère, pas leur contenu.

LE MOT JUSTE

On pourrait appeler **calibration métier** cette étape, encore peu explorée, qui se glisserait *avant* l'indexation. Non pas modifier les documents, mais capter rapidement, avec une intervention humaine légère, la connaissance implicite des experts : marquer les documents critiques, signaler les sources obsolètes, hiérarchiser ce qui fait autorité.

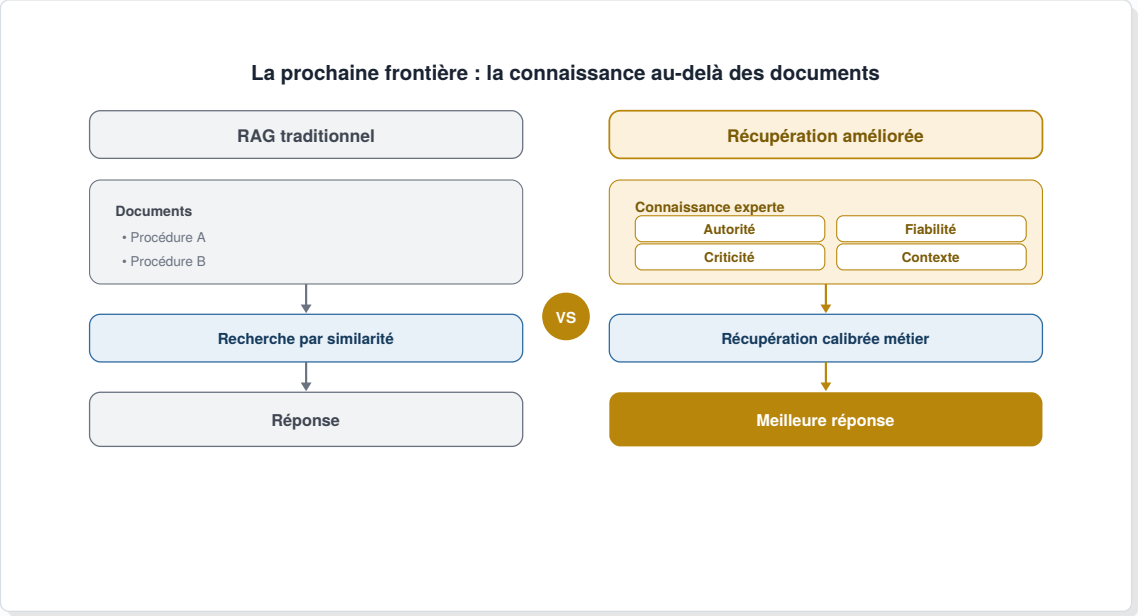


Figure 7. La prochaine frontière. À gauche, le RAG traditionnel ne juge que la ressemblance. À droite, une récupération « calibrée métier » intègre l'autorité, la fiabilité, la criticité et le contexte – la connaissance que les documents ne portent pas d'eux-mêmes.

Le retrieval cesserait alors d'être une simple affaire de ressemblance. À la question « ce passage ressemble-t-il à la demande ? », il ajouterait : « ce passage fait-il autorité ? est-il à jour ? est-ce vraiment celui qu'un expert citerait ? » Chercher ne serait plus seulement comparer des textes, mais aligner une intention métier sur une mémoire documentaire enrichie.

L'IDÉE À RETENIR

L'hypothèse, en une phrase : la qualité d'un RAG dépend autant de la *représentation métier* du corpus que de la finesse de sa recherche. Si elle se vérifie, alors un peu de savoir expert injecté au bon moment vaudrait mieux que beaucoup d'optimisation technique. Le futur du RAG ne serait pas de mieux chercher dans les documents, mais de mieux comprendre comment les experts les utilisent.

Disons-le clairement, pour ne pas faire passer une intuition pour une certitude : nous ne savons pas encore si cette voie s'imposera. Ce n'est pas un concept établi, c'est une hypothèse

de travail – presque un programme de recherche. Mais elle a un mérite : elle répond à une limite que beaucoup de praticiens constatent sur le terrain, sans toujours parvenir à la nommer. C'est souvent ainsi que progressent les idées – en mettant des mots sur une gêne partagée.

Ce qu'il faut retenir

Quelques idées suffisent à tenir l'essentiel. **Un.** Les anciens chatbots imitaient le langage par des règles ; ils étaient aveugles hors de leur script. Les grands modèles, eux, saisissent le *sens* – c'est la rupture qui a tout déclenché. **Deux.** Mais seul, un modèle ne sait que ce qu'il a lu, et invente quand il ignore. Le RAG le rend fiable non en le rendant plus savant, mais en lui donnant les bons documents à lire au moment de répondre – ce qui suppose, comme Google, de savoir les retrouver et les classer. **Trois.** Si tant de projets d'entreprise échouent, ce n'est pas faute d'un cerveau assez puissant, mais faute d'avoir soigné la partie discrète qui choisit quoi lui mettre sous les yeux : découper, ranger, chercher par le sens, trier. Le génie de l'IA en entreprise ne tient pas dans le modèle. Il tient dans le classeur. **Quatre.** Et quand une question dépasse la simple recherche, on confie au modèle l'initiative de chercher en plusieurs fois et de raisonner : c'est l'agent. La même histoire, encore : de la machine qui exécute à la machine qui s'adapte. **Cinq.** Enfin, une piste se dessine : et si l'enjeu n'était plus seulement de mieux *chercher*, mais de mieux *représenter* le savoir métier – autorité, fiabilité, criticité – que les documents ne disent pas d'eux-mêmes ? C'est sans doute la prochaine frontière.

Demain : où va-t-on dans les cinq prochaines années ?

Hasardons, pour finir, un regard vers l'horizon – avec la prudence qu'impose tout exercice de prospective. Les détails techniques changeront, comme ils ont toujours changé. Les modèles seront plus rapides, moins coûteux, plus multimodaux. Mais ces évolutions, aussi spectaculaires soient-elles, ne sont probablement pas les plus importantes. Quelques tendances de fond semblent aujourd'hui assez solides pour être nommées. **Du document isolé vers la mémoire d'entreprise.** La bascule esquissée dans cet article devrait s'accentuer : on cessera de voir le corpus comme un simple ensemble de fichiers à fouiller, pour le concevoir comme une véritable *mémoire structurée*. Une mémoire qui sait ce qui fait autorité, ce qui est à jour, ce qui prévaut en cas de conflit, et comment les connaissances se relient entre elles. La valeur se déplacera progressivement de l'algorithme de recherche vers la qualité de cette mémoire. **Des assistants qui répondent vers des collègues qui agissent.** Les agents quitteront peu à peu le périmètre des tâches les plus simples pour prendre en charge des enchaînements plus longs : préparer un dossier complet, suivre une procédure de bout en bout, coordonner plusieurs outils. Toujours sous supervision, toujours sur le principe « automatiser ce qu'on peut vérifier », mais avec un périmètre qui s'élargira année après année. **De la recherche vers le jugement assisté.** Aujourd'hui, les systèmes excellent à retrouver une information. Demain, ils aideront davantage à l'interpréter. Non pas à remplacer le jugement humain, mais à le préparer : signaler les contradictions, mettre en évidence les précédents pertinents, attirer l'attention sur les exceptions ou les risques. Le défi ne sera plus seulement de trouver le bon document, mais de comprendre pourquoi il compte. **Du savoir explicite vers le savoir implicite.** La prochaine frontière pourrait bien être là. Les entreprises consacrent aujourd'hui

beaucoup d'efforts à numériser leurs documents ; elles consacrent encore peu d'énergie à représenter la façon dont leurs experts utilisent ces documents. Ce savoir implicite – autorité des sources, hiérarchie des règles, exceptions connues, signaux faibles – constitue pourtant une part essentielle de la compétence collective. Les organisations qui parviendront à le capturer et à le transmettre disposeront d'un avantage considérable. **Moins de bricolage, plus de gouvernance.** À mesure que ces systèmes entreront dans les processus réels, la question ne sera plus seulement « est-ce que ça marche ? », mais « qui en est responsable, comment le contrôle-t-on, comment l'audite-t-on ? ». La traçabilité, la maîtrise des coûts, la conformité réglementaire et la capacité à expliquer les décisions deviendront aussi importantes que la performance brute.

L'IDÉE À RETENIR

Le fil rouge des cinq prochaines années ne sera probablement pas une prouesse technique supplémentaire, mais un déplacement du regard. L'avantage n'ira pas nécessairement aux organisations disposant des modèles les plus puissants – ceux-ci deviendront progressivement accessibles à tous – mais à celles qui auront compris que leur véritable actif stratégique est leur savoir, et la façon dont elles l'organisent, le maintiennent et le transmettent. La technologie se banalise. La connaissance, elle, reste profondément singulière.

Ces cinq tendances ne sont pas cinq chantiers séparés : elles convergent toutes vers un même point. Qu'il s'agisse de transformer les documents en mémoire, de confier l'action à des agents, d'assister le jugement, de capturer le savoir implicite ou d'instaurer une gouvernance, c'est toujours la *connaissance de l'entreprise* – sa qualité, sa structure, sa transmission – qui se trouve au centre de gravité. La figure ci-dessous résume ce mouvement d'ensemble.

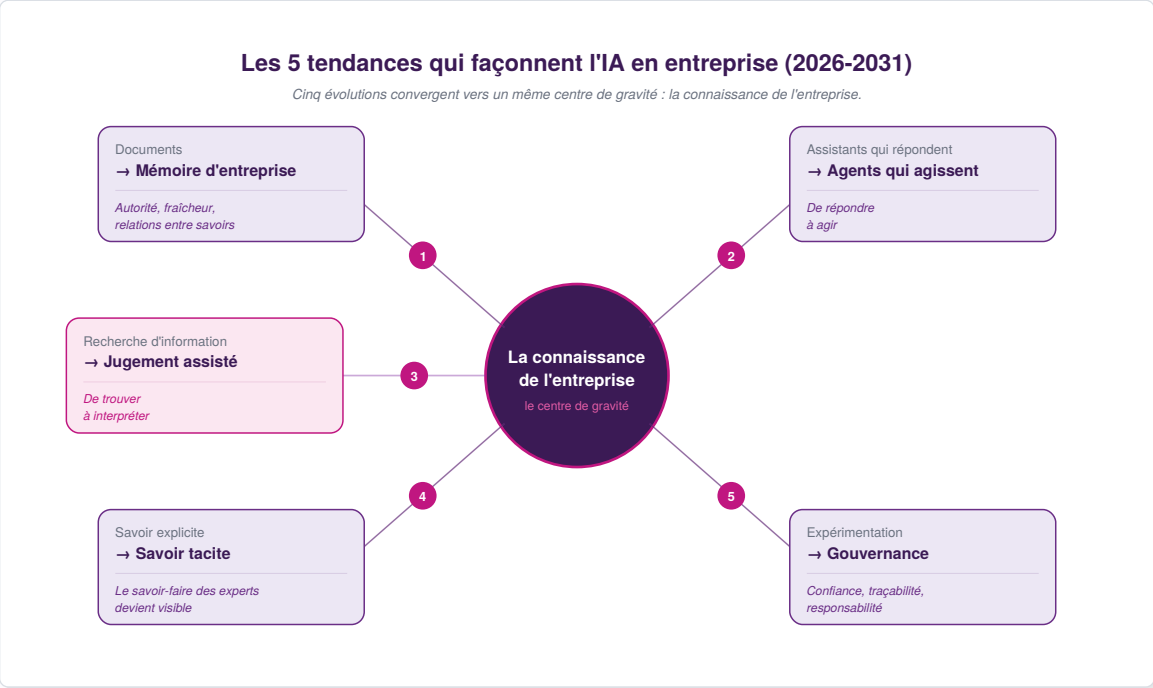


Figure 8. Les cinq tendances qui façonnent l'IA en entreprise pour les années à venir (2026-2031). Toutes convergent vers un même centre de gravité : la connaissance de l'entreprise, sa qualité et la façon dont elle est organisée, maintenue et transmise.

Et c'est peut-être la plus belle nouvelle de cet article. Réussir l'IA en entreprise ne sera pas d'abord une affaire de spécialistes, ni même de budgets colossaux. Ce sera avant tout une affaire de soin apporté à sa propre connaissance : un chantier où l'expertise métier, le bon sens, la rigueur documentaire et la capacité à faire circuler le savoir compteront autant que la technique elle-même. Dans cette perspective, l'intelligence artificielle apparaît moins comme une révolution technologique que comme un révélateur. Elle oblige les organisations à regarder de plus près ce qu'elles savent réellement, comment elles le savent, et comment elles le transmettent. C'est peut-être là que se joue le véritable enjeu des années à venir.