JENNIFER ORIWOH

Technology. People. Decisions.

Researcher and Writer

Selected writing on AI, digital services, software, cybersecurity and accessibility.

01 Before You Give an AI Agent Authority, Decide What It Must Escalate

Executive article · 888 words · Page 2

02 Your Online Form Is Fast. Why Is the Service Still Slow?

Business op-ed · 761 words · Page 4

03 Before a Nigerian Business Buys an AI Platform

Research brief · 974 words · Page 6

04 The Exit Test Every Software Purchase Needs

Business analysis · 764 words · Page 8

05 Stop Making Employees the Last Line of Defence

Leadership commentary · 730 words · Page 10

06 Accessible Design Must Include the Return Journey

Product essay · 787 words · Page 12

07 An AI Answer With Citations Still Needs Checking

Research commentary · 778 words · Page 14

EXECUTIVE ARTICLE / AI & GOVERNANCE

Before You Give an AI Agent Authority, Decide What It Must Escalate

A useful agent needs a clear boundary between making a recommendation and making a commitment.

Jennifer Oriwoh

Consider a hypothetical customer-service assistant handling a refund. It finds the order, reads the policy and prepares a sensible response. The demonstration looks convincing. Then someone connects it to the payment system.

The next question is harder than whether the assistant understands refunds. Can it issue one when the delivery record conflicts with the customer's account? Can it change the destination account? What happens if the payment service accepts the request but the confirmation never arrives?

Those are decisions about authority, evidence and recovery. A successful demonstration does not settle them. Before expanding an agent's access, a business should define the actions it can complete independently, the circumstances that require escalation and the person responsible for resolving an exception.

Separate capability from permission

The word agent can obscure an important design choice. In its 2024 engineering article, Anthropic distinguishes predefined workflows from systems in which a model directs its own steps and tool use. It also describes a tradeoff between flexibility, cost and latency. The article reflects a vendor's engineering experience; it is useful architectural guidance, rather than independent proof of a business return.

That distinction matters to the refund example. A fixed workflow can require an order lookup, an eligibility check and an approval before payment. A more flexible agent might decide which records to consult and when to act. Neither arrangement makes the payment decision harmless. Both need permissions enforced by the surrounding software.

An instruction telling the model to be careful is too weak a foundation for authority. The tool that issues refunds should enforce the permitted amount, eligible transactions and approved destination. Requests outside that scope should be rejected or routed for review, even when the agent supplies an eloquent justification.

Give escalation a destination

An escalation policy should name observable conditions. Conflicting records, missing evidence, a request to bypass a control and an uncertain payment outcome are more useful triggers than a general instruction to escalate difficult cases. Difficulty is subjective; an unresolved discrepancy can be identified and recorded.

The receiving person also needs a usable case. Sending an entire conversation with a request to investigate transfers the workload without organising it. A better handoff includes the proposed action, the records used, the unresolved issue and any action already taken. It should be possible to decide what happens next without repeating the whole investigation.

This creates a management obligation. Someone must own the queue, have authority to resolve it and be available within a stated service window. If no one responds, the system needs a defined waiting state. Silence must not quietly become approval.

The NIST Generative AI Profile treats risk as dependent on the system and its use, within a voluntary risk-management framework. That supports examining the actual workflow rather than assigning an agent a single, reassuring safety label. It does not supply an organisation's escalation thresholds or guarantee a safe deployment.

Test the awkward cases before widening access

An evaluation built entirely from ordinary requests can conceal the cases that make authority expensive. The pilot should include missing orders, duplicate requests, changed account details, unavailable services and contradictory policy documents. Each case needs an expected response before the test begins.

Recovery deserves particular attention. If a payment request times out, repeating it immediately may be inappropriate: the first request could already have succeeded. The system should reconcile the transaction through an authoritative record before deciding to retry. A customer-facing message should distinguish a pending confirmation from a completed refund.

The review should measure more than the proportion of tasks completed. Useful proposed measures include accepted outcomes requiring no correction, exceptions per task, time spent reviewing exceptions and time needed to recover from a failed action. These are evaluation recommendations, not results from a deployment described here.

Reviewers should examine the cases behind an aggregate score. A system that handles routine questions well but mishandles account changes may still be useful within a narrower permission boundary. Expanding access should follow evidence about the newly permitted action, rather than confidence borrowed from unrelated successes.

Avoid turning every action into an approval queue

There is a fair objection: requiring human approval everywhere can remove much of the value of automation. A customer gains little if a supposedly autonomous assistant simply creates another queue.

The answer is to vary authority with consequence and reversibility. Drafting a response, retrieving a record and making a financial commitment deserve different controls. A business can permit routine, bounded actions while reserving unusual or consequential cases for a person. The boundary should be explicit enough to implement and revise.

Nor should a good pilot make that boundary permanent. Policy changes, new tools and altered customer behaviour can invalidate earlier assumptions. The owner needs a way to reduce access, investigate incidents and retest the affected paths. Expanding autonomy should be a reversible operational decision where possible.

The approval worth seeking is therefore specific: this system may perform these actions, using these records, under these conditions. These exceptions go to this owner. These failures suspend the permission.

That is a more demanding brief than asking whether an agent can complete a task. It also gives the business something useful to govern after the demonstration ends.

Sources

Anthropic, Building effective agents.

NIST, Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile, introduction and risk overview.

BUSINESS OP-ED / DIGITAL SERVICES

Your Online Form Is Fast. Why Is the Service Still Slow?

The most revealing measure of a digital service may begin where its interface ends.

Jennifer Oriwoh

Imagine a customer completing an online request in two minutes. The form is clear, the confirmation arrives immediately and the dashboard records another successful submission. The request then waits for verification, moves to an approval queue and returns because one department needs information another has already collected.

This is an illustrative scenario, not a measured company case. It exposes a question that a launch dashboard can miss: when does the customer actually receive the service?

A business can improve submission without improving resolution. That does not make the new form worthless. It means the investment has solved one part of the customer's problem, and its success should be described accordingly.

Follow the request beyond the screen

An interface is easy to demonstrate. The work behind it is harder to put on a presentation slide. Someone must verify the information, make a decision, communicate it and deal with exceptions. Each stage can have a different owner, system and definition of completion.

The danger appears at the boundary. A team can finish its own task while the request remains unresolved. Sending a document counts as a completed handoff for the sender; receiving an incomplete document creates new work for the recipient. Both teams may meet their internal targets while the customer keeps waiting.

The GOV.UK Service Standard's guidance on solving a whole problem asks teams to design around users' needs and agree how their work joins up with other organisations. It also warns against trying to solve everything at once. Although written for public services, that combination offers a useful discipline for commercial teams: understand the whole journey, then improve a manageable part of it.

For a business, that could mean tracing a request from arrival to a correct decision. The exercise should include returned work, offline checks and calls made by the customer to find out what is happening. A process map that excludes those activities describes the intended service more neatly than the experienced one.

Put waiting time in the same picture as working time

The distinction between processing and waiting changes the investment conversation. If staff handle a request quickly once they receive it, another tool to accelerate data entry may have limited effect on the overall delay. The larger opportunity could be a queue reviewed too infrequently, an unclear approval limit or a missing rule for incomplete cases.

That is a hypothesis to investigate, not a reason to assume that every approval is waste. Some checks protect customers, prevent fraud or satisfy genuine obligations. Removing them without understanding their purpose can produce faster errors.

The useful question is what each wait achieves. Does it enable a necessary judgment? Is the required information available? Does anyone know the request is waiting? A control can remain necessary even when the way work reaches it needs redesign.

GOV.UK's performance guidance connects measures to whether a service solves its intended problem and calls for online and offline data alongside user research. Applied cautiously to a business, that suggests measuring total time to a correct resolution, returned requests and customer follow-up, as well as the speed of individual tasks.

The measures should expose the slowest cases too. An average can improve while a smaller group continues to experience long, unexplained waits. Separating routine cases from exceptions can show where the service needs a different response rather than more pressure to work faster.

Give the handoff an owner

Improvement does not always require replacing the entire system. A bounded review can select one request type, identify its longest avoidable wait and agree who owns the next step. The receiving team should help define what a complete handoff contains and how rejected work returns to the sender.

Customer communication belongs in the same review. A confirmation that merely says the form was received may be accurate, but it leaves the next event unclear. A useful update explains what happens next, whether the customer needs to act and how an unresolved delay will be handled. It should not promise a deadline the service cannot support.

Before buying another layer of software, the operations leader should ask the team to show one request travelling through the existing process. Where did it stop? Who noticed? Who could move it forward?

The next digital investment should answer the problem that exercise reveals. A faster form is a worthwhile improvement. A service earns the customer's confidence when the work continues after they press Submit.

Sources

Government Digital Service, Solve a whole problem for users.

Government Digital Service, Define what success looks like and publish performance data.

RESEARCH BRIEF / TECHNOLOGY BUYING

Before a Nigerian Business Buys an AI Platform

A purchasing brief should test the operating conditions behind the demonstration.

Jennifer Oriwoh

The most useful first question in an AI purchasing meeting is what the business needs to improve. A platform can produce an impressive answer while leaving that question unresolved.

Consider a hypothetical Lagos service business comparing tools for handling customer enquiries. It wants staff to spend less time finding approved information and more time resolving requests. The decision is whether a particular change can improve that work at an acceptable cost and level of risk. Buying AI is one possible answer, alongside better records, ordinary automation and a more organised current process.

The Nigerian setting gives the buyer a concrete place to investigate data handling, payment arrangements, support and connectivity. It does not justify assuming that all Nigerian businesses face the same constraints, or that an overseas product will necessarily fail. A purchasing recommendation should change when the business's circumstances change.

Start with the information the system will receive

For the enquiry use case, assemble an authorised test set that reflects the work: current policies, superseded documents, incomplete questions and requests requiring a person. Record what a correct answer should contain and when the system should decline to answer. Keep private customer records out of an initial demonstration unless their use has been properly authorised and protected.

Then test whether the tool identifies which information is current. An answer that correctly summarises an obsolete policy can still be wrong for today's customer. A source link helps a reviewer investigate; it does not by itself prove that the answer follows the source or that the source applies.

Language and context should be tested through the customers and tasks the business actually serves. If enquiries include multiple languages, informal expressions or sector-specific terms, those belong in the evaluation. Broad claims about African language support cannot replace examples relevant to the intended users.

The World Bank's *Digital Progress and Trends Report 2025* organises AI foundations around connectivity, compute, context and competency. Its discussion of locally relevant data and skills offers a useful starting lens. It is a development-level framework, not evidence that a particular platform will repay this business's investment.

Compare four credible options

Keeping the current process is a legitimate baseline. Measure the time, errors and unresolved requests it produces before claiming that an alternative improves them. The baseline may itself benefit from clearer responsibilities or a better-maintained knowledge base.

Ordinary automation is the second option. If enquiries fall into stable categories with approved responses, better search, templates or routing may address much of the problem. That option deserves a fair trial rather than being dismissed because it sounds less ambitious.

A purchased AI service may offer flexible language handling with less development work. The buyer should establish what it includes, which work remains with staff and how changes to the service will be communicated. A subscription price is only one part of that comparison.

A limited custom component offers another route when a specific integration or control is essential. It also creates maintenance, monitoring and support responsibilities. Building something small does not remove the need for someone to own it after launch.

Calculate the cost of accepted work

For each option, estimate setup, integration, subscription or usage charges, staff review, training, support and likely exit work. Where billing is in a foreign currency, model how a change in the naira cost would affect the business's budget. Use current supplier terms for the calculation rather than assuming an exchange rate or a fixed price here.

The denominator matters. Dividing expenditure by every generated answer rewards volume even when answers need correction. A more useful proposed measure is cost per accepted resolution, with a clear definition of accepted and a separate record of serious errors.

Connectivity needs the same practical treatment. Test the service on the connections and devices staff will use. Interrupt a permitted test request and establish whether work is saved, duplicated or lost. Ask what staff should do while the service is unavailable. These tests address the actual operating environment without turning a national infrastructure assumption into a product verdict.

Establish who controls the data and the decision

Before uploading operational records, ask the supplier to explain where data goes, how long it remains, who can access it and whether it is used to train models. Obtain answers that apply to the service tier and contract under consideration. A generic privacy statement may not settle the specific arrangement.

The business should have its responsible data-protection adviser review the proposed processing against applicable Nigerian requirements before using personal data. This brief does not determine the legal basis for a particular deployment. Commercial convenience is not a substitute for resolving that question.

Operational ownership also needs a name. Someone must maintain the source material, review disputed answers and decide when the tool should be suspended. Ask the supplier to demonstrate export and deletion options using permitted test data. An exit promise is more useful when the buyer understands what can actually be recovered.

Make the pilot capable of saying no

Agree the acceptance conditions before the trial: the quality improvement required, the affordable operating cost, the errors that block rollout and the fallback if the tool is withdrawn. Include ordinary staff in the test rather than relying entirely on a vendor-led demonstration.

A small pilot cannot establish every risk or seasonal demand pattern. It can reveal whether the purchasing case has survived contact with real work. If staff spend more time correcting answers than the tool saves, narrow the use case or stop. If a simpler option performs adequately at lower total cost, buying less technology is a defensible outcome.

The purchase should follow a demonstrated improvement in a defined process. The ability to produce an impressive answer is the beginning of that assessment, not its conclusion.

Source

World Bank, Digital Progress and Trends Report 2025: Strengthening AI Foundations, official report overview.

BUSINESS ANALYSIS / CLOUD & SOFTWARE

The Exit Test Every Software Purchase Needs

Dependence on a supplier can be a sensible choice. It should be a choice the buyer understands.

Jennifer Oriwoh

A cancellation clause can make a software contract look reassuringly flexible. It says when payment can stop. It says much less about how the business will continue working afterwards.

By the time a tool becomes useful, it may hold records, approval rules, integrations and knowledge about how work gets done. Those dependencies can outlast the subscription. The practical exit question is whether the organisation can recover enough of them to operate somewhere else.

That question belongs in the buying process. Once a service is essential, discovering that its exports omit attachments or that no one understands the integration becomes a much more awkward negotiation.

Understand the dependence you are accepting

UK government guidance on technical lock-in makes an important distinction between contractual restrictions and technical dependence. It also recognises that accepting some dependence can deliver benefits, including simpler services and less operational work. Avoiding every proprietary feature can impose costs of its own.

That is a better basis for discussion than insisting that all lock-in is bad. A managed service may be valuable precisely because the supplier takes on difficult work. Rebuilding those capabilities internally simply to preserve an abstract ability to leave could consume resources the business needs elsewhere.

The judgment is whether the benefit justifies the likely cost of changing direction. A commodity tool and a service central to fulfilling customer commitments may deserve different exit arrangements. Criticality, available alternatives and the organisation's capacity to manage a move should shape the decision.

For software buyers outside government, this is a commercial application of the guidance, not a claim that its public-sector rules apply to every purchase.

Ask for a rehearsal, not a promise

A small exit test can reveal the difference between exportable data and usable work. Create a permitted sample containing the kinds of records the business expects to retain: a completed request, an attachment, a comment, a status change and a relationship between records. Export it through the route the supplier offers.

Then inspect the result away from the original interface. Can another person identify each record, read the attachment and understand which items belong together? Are timestamps and status meanings clear? If the export contains codes, is the information needed to interpret them available?

This does not require a complete migration during procurement. It requires enough evidence to understand the boundary. A supplier may export the records while leaving workflow configuration or integration behaviour behind. That may be acceptable, but the buyer should know what would need to be rebuilt.

The test should include the receiving side. A file that opens successfully is not proof that the next system can import it or preserve its meaning. For important services, identify a credible destination or a temporary operating method and establish which transformations would be required.

Budget for the people and the transition

The cost of leaving includes more than downloading files. Staff may need to reconcile old and new records, learn a changed process and answer customer questions during the transition. Integrations must point to the right destination. Access and retention decisions need to be handled deliberately.

Procurement should establish who can perform the export, how long access remains available and what assistance is included. Charges, notice periods and deletion arrangements should be checked against the actual contract. A demonstration account cannot establish the rights attached to a later commercial agreement.

The organisation also needs its own record of how the service is used. If the only person who understands a critical integration leaves, supplier flexibility will not solve the knowledge gap. Keeping configuration notes and naming an internal owner are modest steps compared with rebuilding an undocumented process under pressure.

There is a limit to rehearsal. A small dataset will not establish the duration or reliability of moving the full production workload. The larger and more consequential the service becomes, the stronger the evidence required. Repeat the review when usage or dependencies change materially.

Buy the benefit with the dependency visible

A useful purchasing decision might deliberately accept a difficult exit because the service offers substantial value and the alternatives are worse. Another might choose a simpler tool because the organisation cannot support the dependency. Both can be reasonable.

What weakens the decision is treating the ability to cancel payment as proof of operational freedom. Before signing, ask for a small, interpretable export and an honest account of what it leaves behind. The answer will improve the purchase even if the business never leaves.

Source

Government Digital Service and Central Digital and Data Office, Managing technical lock-in in the cloud.

LEADERSHIP COMMENTARY / CYBERSECURITY

Stop Making Employees the Last Line of Defence

Security training matters. So does designing a workplace in which one mistake need not become a crisis.

Jennifer Oriwoh

An employee receives an urgent request that appears to come from a senior colleague. The message concerns a payment, arrives during a busy afternoon and sounds plausible. In this hypothetical case, the employee acts before checking it independently.

The easy response is another reminder to be vigilant. The more useful response examines why an unverified message could move money, which account permissions made the action possible and how quickly the organisation could detect and reverse it.

Staff judgment belongs in a security programme. It should not have to compensate for every weakness in the surrounding process.

Give caution a workable process

The UK National Cyber Security Centre's guidance on spotting cyber attacks describes phishing through emails, texts and calls that pressure people into disclosing information or transferring money. It advises verifying doubtful messages through independently obtained contact details. That is practical advice; it becomes more useful when the business makes it easy to follow.

For a payment change, the process should explain who verifies the request, which trusted contact route to use and what happens when the requester cannot be reached. Finding a phone number inside the suspicious message defeats the purpose. So does a workplace in which delaying an executive's request is treated as disobedience.

The control also needs to survive a busy day. A procedure that requires lengthy investigation for every routine transaction is likely to create queues and workarounds. Reserve stronger checks for consequential changes, make the verification route clear and give staff authority to pause an uncertain request.

Training can then use the actual workflow rather than a generic warning about suspicious spelling. The relevant lesson is how to respond to a plausible request under pressure, including when a familiar-looking sender is insufficient evidence.

Protect accounts as well as attention

Account security provides another layer. The NCSC's guidance on recommended authentication methods prioritises FIDO2 credentials where appropriate and distinguishes their phishing resistance from weaker methods. Its recommendations consider the available devices and the organisation's requirements. Not every form of multi-factor authentication provides the same protection.

A business should therefore check which methods its important services actually support, how administrators enrol users and how a lost device is handled. Enabling a stronger login while leaving recovery poorly controlled can undermine the intended protection. Staff also need a usable route back into their accounts.

Access should reflect the task. A person who needs to read a customer record may not need to export every record or change payment details. Reducing unnecessary permissions cannot prevent every incident, but it can reduce what a compromised account is able to do. The permissions review should include people who have changed roles or left.

These decisions require cooperation between operations and technical staff. A security setting that blocks a legitimate task without a supported alternative can encourage people to move the work into less controlled channels.

Prepare to recover without improvising

Prevention will never answer every operational question. If staff lose access to important records, the business needs to know what can be restored and who can authorise the recovery.

NCSC backup guidance recommends copying the data an organisation needs to operate, protecting those copies and knowing how to restore them. A completed backup notification is a useful signal, but a restore exercise provides different evidence: whether the records can be recovered and used.

A proportionate rehearsal can restore permitted test data into an isolated destination, confirm that important files are present and record the steps that were unclear. It should not overwrite live work merely to prove that a backup exists. The result can inform realistic expectations about disruption without pretending that a small rehearsal certifies the whole recovery plan.

Reporting is part of recovery too. Someone who realises they may have approved a fraudulent request should know where to report it immediately. If the predictable response is humiliation, the organisation risks making early disclosure harder. Managers should distinguish a deliberate breach of rules from an honest report of a mistake.

The purpose of these measures is to give vigilance support: a verification route, stronger account protection, limited authority and a practiced response. A business that wants careful staff should build conditions in which being careful is possible, and in which a lapse can be contained.

Sources

NCSC, Spotting cyber attacks.

NCSC, Recommended types of MFA.

NCSC, Backing up your data.

PRODUCT ESSAY / ACCESSIBILITY

Accessible Design Must Include the Return Journey

Getting through the first screen is only part of being able to use a service.

Jennifer Oriwoh

A person can read a website, navigate its menus and still be unable to complete the task that brought them there. The obstacle may appear when they log in again, correct a form or recover an account. Accessibility has to reach those moments too.

That matters because a service is rarely one uninterrupted visit. People stop to find a document, switch devices, make an error or return after an absence. A design review that follows only the cleanest path can miss the points where access becomes difficult.

The return journey deserves its own attention: what happens when a person needs to resume, correct or recover? It is a practical way to examine the assumptions hidden inside familiar interfaces.

Memory should not be an unnecessary entry requirement

Login is an obvious place to start. W3C's explanation of WCAG 2.2's Accessible Authentication criterion addresses cognitive tests in authentication, including remembering passwords and transcribing information. The criterion permits specified alternatives or assistance, and includes exceptions for object recognition and personal content. Supporting password managers and copy-and-paste are examples of mechanisms that can reduce the burden.

The distinction is important. Accessibility does not require abandoning account security or making every login identical. It requires examining whether people have an appropriate way to authenticate without an avoidable cognitive barrier. The criterion concerns existing-account authentication, rather than setting every rule for account creation.

A team can begin with ordinary tests: does a password manager recognise the fields? Can someone paste a verification code? Does the interface accept the complete pasted value? A system that displays several code boxes may look tidy while behaving badly when someone tries to paste into them.

Account recovery belongs in that review. An accessible primary login loses much of its value if the only recovery route depends on a task the person cannot perform. Review the path as a connected process, including the external service used to verify identity.

Preserve useful information without keeping everything

Forms create another kind of memory demand. WCAG 2.2's Redundant Entry criterion generally requires information already provided in the same process to be populated or available to select when requested again. It includes exceptions for essential re-entry, security and information that is no longer valid. It does not require keeping information between sessions.

That last qualification matters to the broader return-journey argument. There may be good reasons to let someone resume later, but this particular criterion is not a mandate to store their form indefinitely. Any persistence feature needs a deliberate decision about privacy, access and retention.

Within a process, the practical goal is to avoid making someone reconstruct information the service already has. A review screen should let them inspect and change the relevant answer. An error in one field should

not create needless work elsewhere. The design should make clear what remains saved and what must be entered again.

Consider an illustrative application form in which a person corrects a reference number. If the entire form clears, the apparent correction task becomes a second application. That additional work can be especially difficult for someone using assistive technology or managing fatigue. A short field-level error message does not solve the larger problem if the process itself discards progress.

Test recovery with the people who need it

A useful review deliberately leaves the ideal path. Enter an invalid value, move back a step, request a new code and try an expired link. Establish whether the interface explains the problem, identifies a next action and preserves information where appropriate.

Those checks should include keyboard use and relevant assistive technologies, with disabled participants involved in research and testing. A designer's inspection can find defects, but it cannot establish how every person will experience the service. Automated checks are valuable within their coverage; they do not replace that inquiry.

There are real tradeoffs. Keeping progress can help someone resume while creating exposure on a shared device. A longer session may support a person who needs more time while increasing the period in which an unattended account is accessible. Teams should examine those conditions rather than treating either convenience or security as an automatic answer.

The recommendations here extend beyond two selected success criteria. Following them does not establish full WCAG conformance, which requires assessment of all applicable requirements. Their value is to direct attention to a neglected part of the product conversation.

Before approving a release, ask the team to demonstrate a mistake and a return visit, as well as a successful first attempt. A service becomes more usable when people can find their way through the ordinary interruptions of using it.

Sources

W3C Web Accessibility Initiative, Understanding SC 3.3.8: Accessible Authentication (Minimum).

W3C Web Accessibility Initiative, Understanding SC 3.3.7: Redundant Entry.

RESEARCH COMMENTARY / AI & INFORMATION

An AI Answer With Citations Still Needs Checking

A reference is valuable when a reader can follow it to evidence that supports the claim.

Jennifer Oriwoh

A link beside a confident sentence changes how an answer feels. It suggests that someone has done the checking. For a busy reader using an AI-generated research summary, that suggestion can be enough to move the sentence into a presentation, a proposal or a decision.

Yet several different things must be true before the reference earns that trust. The source must exist. It must contain the relevant evidence. The answer must represent that evidence accurately, and the evidence must apply to the question being asked.

Treating those as separate checks makes a source trail more useful. It also reveals why simply asking an AI system to include citations does not finish the research.

A real source can support the wrong sentence

Consider a hypothetical summary claiming that a tool improves productivity across small businesses. Its citation leads to a genuine study of a specific task, among a particular group, under controlled conditions. The link works. The broad claim still goes beyond the evidence.

The mistake may be a changed population, a missing comparison or a stronger verb than the study permits. A report that documents an association cannot automatically support a claim that one factor caused another. A vendor's account of a successful customer does not establish what a typical buyer should expect.

These are familiar research problems, now appearing inside a fast, convenient interface. Checking them requires reading the source rather than inspecting the neatness of the answer. The original date, location, sample and method may matter more than the number of links displayed.

NIST's Generative AI Profile identifies confidently stated false content as a risk and discusses fabricated supporting citations. Its treatment explains why presentation alone is weak evidence of correctness. It does not provide an error rate for every current tool or eliminate the need to test a particular use case.

Measure support, not the appearance of research

The distinction has also been examined in language-model research. In the 2023 paper *Enabling Large Language Models to Generate Text with Citations*, Tianyu Gao and colleagues introduced ALCE, a benchmark that evaluates fluency, correctness and citation quality separately. Its citation measures examine whether statements are supported and whether cited passages are relevant to that support.

That separation is the useful lesson here. A readable answer can fail a support check. A correct statement can have a poor citation. Counting references alone cannot distinguish those outcomes. The paper's experiments concern the systems and datasets studied in 2023; they should not be presented as a measurement of every product available today.

For an organisation evaluating a research assistant, the test should reflect the documents it will actually use. Include a current policy, an older version, an ambiguous request and a question the documents cannot answer. Decide what adequate support looks like before reviewing the generated responses.

Then inspect the important claims individually. Does the cited passage contain the information? Is an exception omitted? Does the answer combine separate facts into a conclusion the source never made? Record an unsupported claim even if another part of the answer is useful.

Make uncertainty usable

A research tool should help a reviewer see where its evidence ends. When documents disagree, an answer that identifies the disagreement may be more useful than one that silently chooses a side. When the answer depends on information outside the available material, a clear limitation protects the next decision.

This does not mean requiring exhaustive verification of every low-stakes sentence. Review effort should match consequence. A rough list of topics for further reading can tolerate a different process from a customer promise, a procurement recommendation or a public statement attributed to an executive.

For consequential work, a compact claim record can help. Keep the proposed statement, the original source, the relevant passage, its date and any qualification needed. The record should distinguish what the source reports from the interpretation the writer adds. It gives an editor a practical route to review the argument without starting from a pile of undifferentiated links.

Access matters as well. A reader needs to know when a source is unavailable or only an abstract has been consulted. A summary should not imply that a full study was examined when the evidence available was a short description. Source transparency includes the limits of the writer's own access.

The strongest use of an AI research assistant is to make evidence easier to find and examine. Its speed becomes valuable when the saved time supports better judgment, rather than allowing an unsupported claim to travel further before anyone opens the link.

Sources

NIST, Generative Artificial Intelligence Profile, section 2.2, Confabulation.

Gao, Yen, Yu and Chen, Enabling Large Language Models to Generate Text with Citations, EMNLP 2023, sections 2-3.