Zeno

SEPTEMBER 2026

ENGINEERING NOTE 01

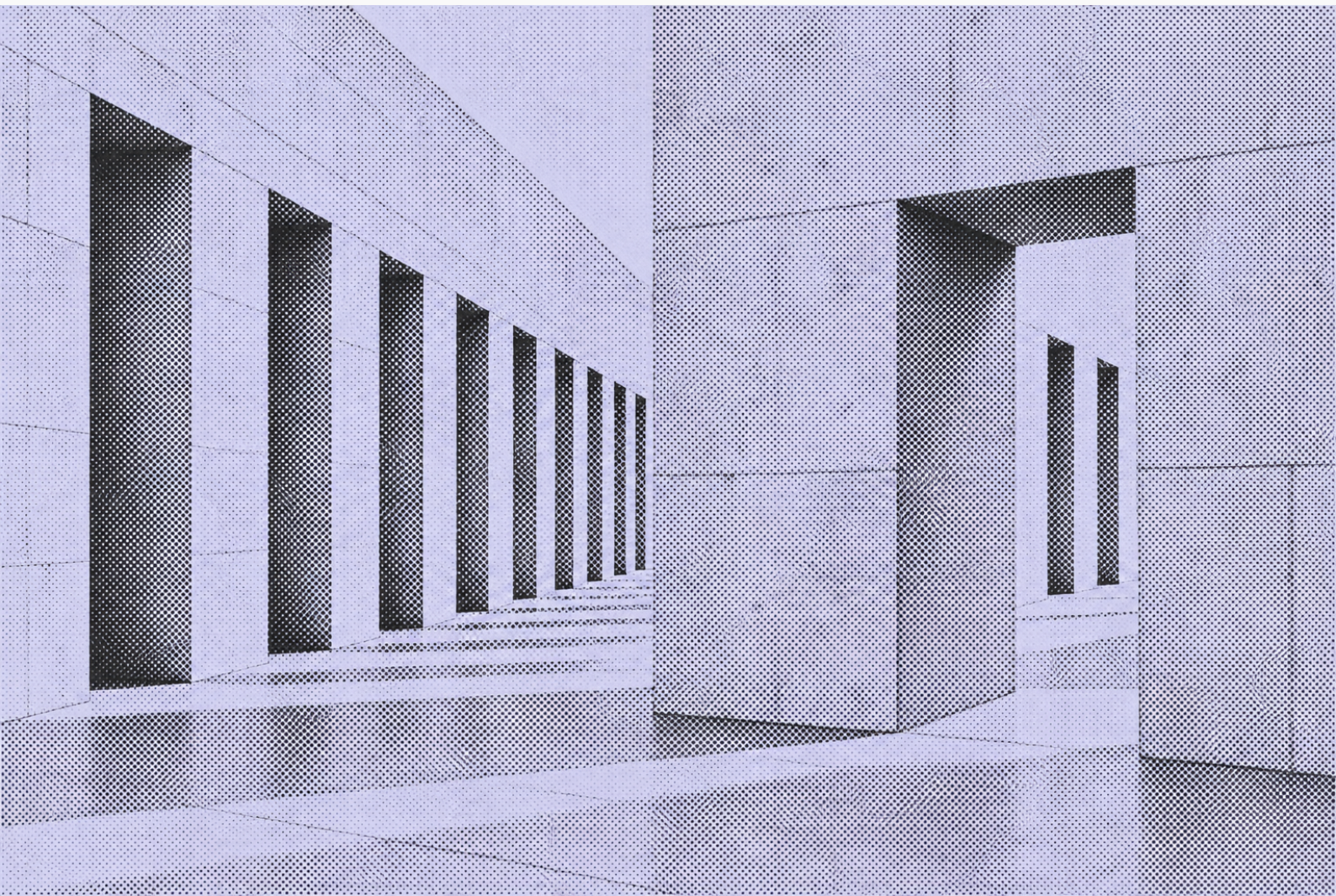
The hallucination trap:

Why a real citation can still lead to the wrong legal answer.

Samantha Swift

Product engineering, Zeno

Article one · Finding the evidence



THE PROBLEM

A real citation is only the beginning

The uncertainty was all-consuming. We were buying a home in the Netherlands when we discovered conflicting information about the condition of its foundations. I wanted to understand whether we could rely on what the seller's agent had told us, what the seller should have disclosed, and what we should have investigated ourselves.

As an Australian, I didn't have the same familiarity with the legal system here. Our home and potentially significant future costs were at stake. I was trying to understand our position and prepare better questions for our lawyer.

Seeing the sources in Zeno made it feel more concrete. I could open the citation viewer and read actual cases to compare with my situation. I was conscious that a reassuring answer might also be the answer I wanted to hear. Being able to inspect the sources helped me ground myself, step back from the emotion and look more critically at what the answer was saying.

I lead Zeno's product engineering team. During that period, I experienced our engineering decisions from the other side of the screen. A reference to a real case gave me something to examine. I still needed to understand whether that case supported the answer to my question.

Hallucinations and other ways an answer can go wrong

"Hallucination" often brings to mind a made-up case or quotation. However, we also need to consider answers that contain real sources but faulty reasoning. Table 1 on page 4 shows several ways an AI answer can go wrong.

Some are hallucinations in the sense of fabricated or unsupported content. Others involve incomplete research or errors in applying the evidence. As engineers, we need that distinction. A check that catches an invented reference will not necessarily catch a missing exception in the law.

BOX 1 / TECHNICAL BACKGROUND

Hallucinations, in plain language

What does 'hallucination' mean?

Generative AI can present information as factual even when it is invented, incorrect or unsupported by the material it is using. For example, an AI meeting summary might say a decision was agreed when people only discussed it. The sentence can sound entirely convincing.[1, 2]

Why does this happen?

A large language model learns patterns from vast amounts of text and generates a response a piece at a time. Producing plausible language is not the same as verifying facts. When information is missing, it can fill the gap with something that sounds likely.

It can also misrepresent a document you give it. Training and tests that reward answering rather than admitting uncertainty can reinforce this tendency. Retrieving reliable sources and checking claims can reduce the risk, but do not eliminate it.[1, 2, 3]

Where did the term come from?

The term predates today's chatbots. An early documented use is Simon Baker and Takeo Kanade's *Hallucinating Faces* (2000), which described generating details in low-resolution images. That use concerned image reconstruction, rather than incorrect chatbot answers.[4]

THE EVIDENCE ON SCREEN

A source you can inspect

Addressing hallucinations throughout the system

At Zeno, we address hallucinations throughout the system. We address citation reliability at several stages:

- Guiding the model's research
- Controlling how the AI tools return sources
- Connecting citations to sources
- Assessing whether those sources support the statements in an answer

This is how we treat hallucination risk as an architectural concern. We give different parts of the system specific responsibilities, rather than expect one instruction to settle the problem.

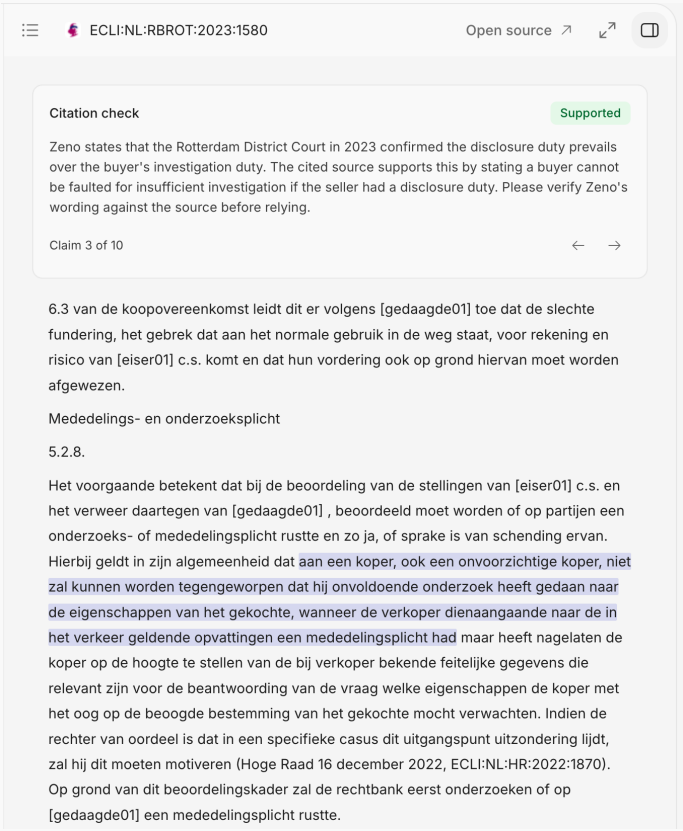


Figure 1. A disclosure and investigation claim appears with a supported assessment and a highlighted passage in ECLI:NL:RBROT:2023:1580. The source remains available for inspection; the assessment does not decide how the rule applies to a particular purchase.

Open the cited judgment

THE RESEARCH PROCESS

Finding the part that matters

Across these two articles, we follow the process from finding evidence to checking its use. This first article covers research and passage selection. The second examines the citation checker, including the possibility that the checker itself gets something wrong.

Imagine preparing a research folder for a colleague. You choose judgments, mark relevant passages and add references worth following. Your colleague uses that material to prepare an answer. Zeno's research process has a similar division of work.

First, the tools find candidate sources. If you know a case reference or an article number, the system can use a lookup tool. If you describe a problem, it needs to search for relevant material.

From a list of candidate sources, we then select which to pass back to the answering model. A judgment may discuss the parties' arguments, the facts, procedural questions and several points of law.

For longer law sources, Zeno uses an AI model to select the relevant sections. That selector identifies portions of the text to be added to the candidate list, but we keep the connections to the original source, so nothing is paraphrased.

The folder analogy also explains the risk. A colleague who receives three marked paragraphs may miss a qualification on another page. The answering model can face the same problem if the selected passages leave out something important.

Returning the whole judgment is not a universal solution either. Models have limits on how much text they can process, and we have to decide how to use that space across several sources. Evidence selection means balancing detail, breadth and relevance.

The wording of the research question matters here. A request about statements made before a sale may lead to different passages from a request about investigations carried out by the buyer. Follow-up questions can help explore an aspect the first search did not cover.

Table 1. Where an answer can go wrong

FAILURE TYPE	EXAMPLE IN A HOUSE-PURCHASE QUESTION
Invented source or quotation	A nonexistent judgment or words the court did not use.
Misrepresented source	An omitted qualification, or a buyer's argument presented as the court's finding.
Wrong context	An overlooked difference in the legal framework, date or circumstances.
Assumed facts	An assumption that the buyer received a report before signing, although the user never said so.
Unsupported conclusion	A buyer succeeded in a cited case, so the answer concludes that this buyer should succeed too.
Missing material	The answer discusses the seller's information but overlooks material relevant to the buyer's investigation.

CONNECTIONS AND COVERAGE

Following the connections

A lawyer reading a judgment may follow a reference to another case or to a statutory provision. Our knowledge graph gives Zeno ways to explore recorded connections between legal materials.

The knowledge graph includes routes for finding cases that cite a previously retrieved authority, attaching related rulings and exploring legislative revision history. These relationships can lead the language model towards material beyond its first search results.

In the research-folder analogy, these are the references your colleague writes in the margin: “read this next”.

The model still needs the underlying text. A court might cite an earlier case to follow it, distinguish it or discuss a separate issue. A connection in the graph helps locate material; assessing its legal significance requires further reading.

For my own question, the useful comparison was more detailed than “another buyer had a problem with a house”. I wanted to understand what information the parties exchanged, what they knew and how those circumstances related to my situation. Those were questions to explore with the sources and my lawyer.

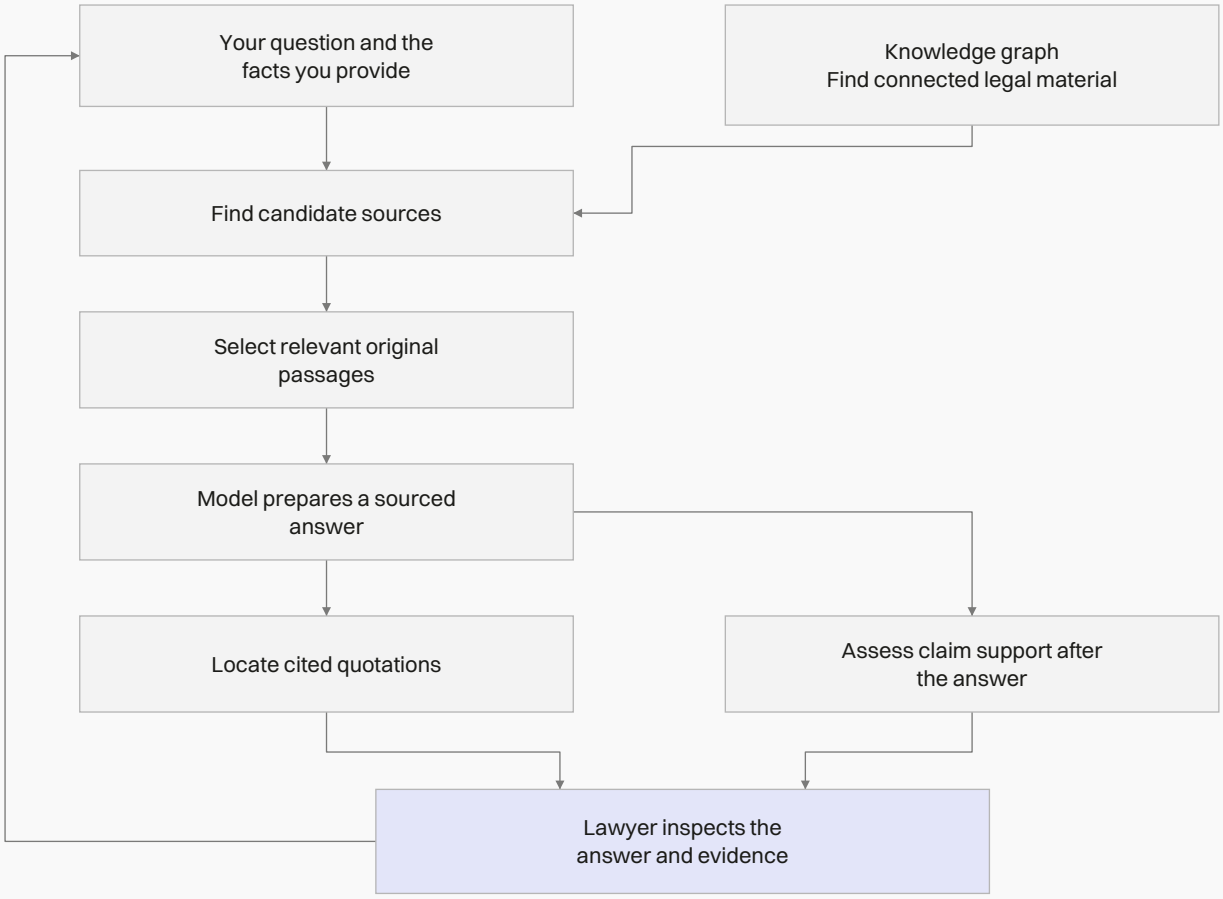


Figure 2. Simplified process. Instructions guide research and generation. Software handles specific constraints; models assess relevance and meaning. Review can lead back to a new question.

LEARNING FROM FAILED SEARCHES

Learning from the sources Zeno could not find

We originally started recording unsuccessful searches to answer a practical question: what legal sources were users asking for that we did not have? Those requests, together with feedback from users, could help us decide where to expand the knowledge graph.

Then we noticed something unexpected. Some of the reported “gaps” in the knowledge graph concerned legislation we KNEW we already held. In the research-folder analogy, our colleague had come back empty-handed even though the material was in the library. We needed to understand what they had asked for and why the search had failed.

So, for certain legislation lookups, we added a separate database check: did we recognise the law’s title, even though we could not retrieve the requested article? This allowed us to separate true gaps in the sources vs “hallucinations” where the model searched for a source we had but still returned no results.

This then gives us a starting point for investigating the failure. For example, the model might have combined the right law with the wrong article number. It might have used a name our lookup did not recognise, or requested an older provision when the question required current law. We also need to check whether our own retrieval tools, filters or source coverage caused the problem.

This discovery completely changed how we treated this new tool. Instead of gap analysis, it turned into “hallucination” hunting our retrieval tools. Adding more sources will not fix a request for the wrong article, the same as changing a prompt will not supply a document we do not have.

Dylan, who contributed to this work, described one pattern we see: a model can have a strong preference for a familiar legal reference it read during training. That familiarity can persist after the law changes. We need to guide our AI towards the provision relevant to the question and its date. An older provision may still matter for a historical question or under transitional rules. One example in Dutch legislation is Article 843a Rv. Article 843a Rv was repealed with effect from 1 January 2025, but is often referenced in historical legislation, so our tools consistently try to use it instead of being referenced to the correct article which IS in force.

For a lawyer, the consequence is straightforward. If a lookup fails, the model has not obtained the evidence it sought through that search, and the user won’t get an answer grounded in that legal source.

LEGAL CONTEXT / ARTICLE 843a Rv

Article 843a Rv was repealed as part of the evidence-law reform effective 1 January 2025. Transitional rules preserve the former law for proceedings already pending at that instance. A reference to Article 843a is therefore not, by itself, evidence of a hallucination.[6]

6. Amending Act, Staatsblad 2024, 62 (Article I, X; Article XIIA) · Commencement decree, Staatsblad 2024, 72

THE INVESTIGATION CYCLE

Investigating the failed search

Recording those failures gives us examples to revisit after changes to the tools or instructions. We can test whether a recurring problem has improved and watch for it returning. When comparing results over time, we also need to account for how often the tools are used and what users are asking. More unsuccessful searches alone do not establish a higher failure rate.

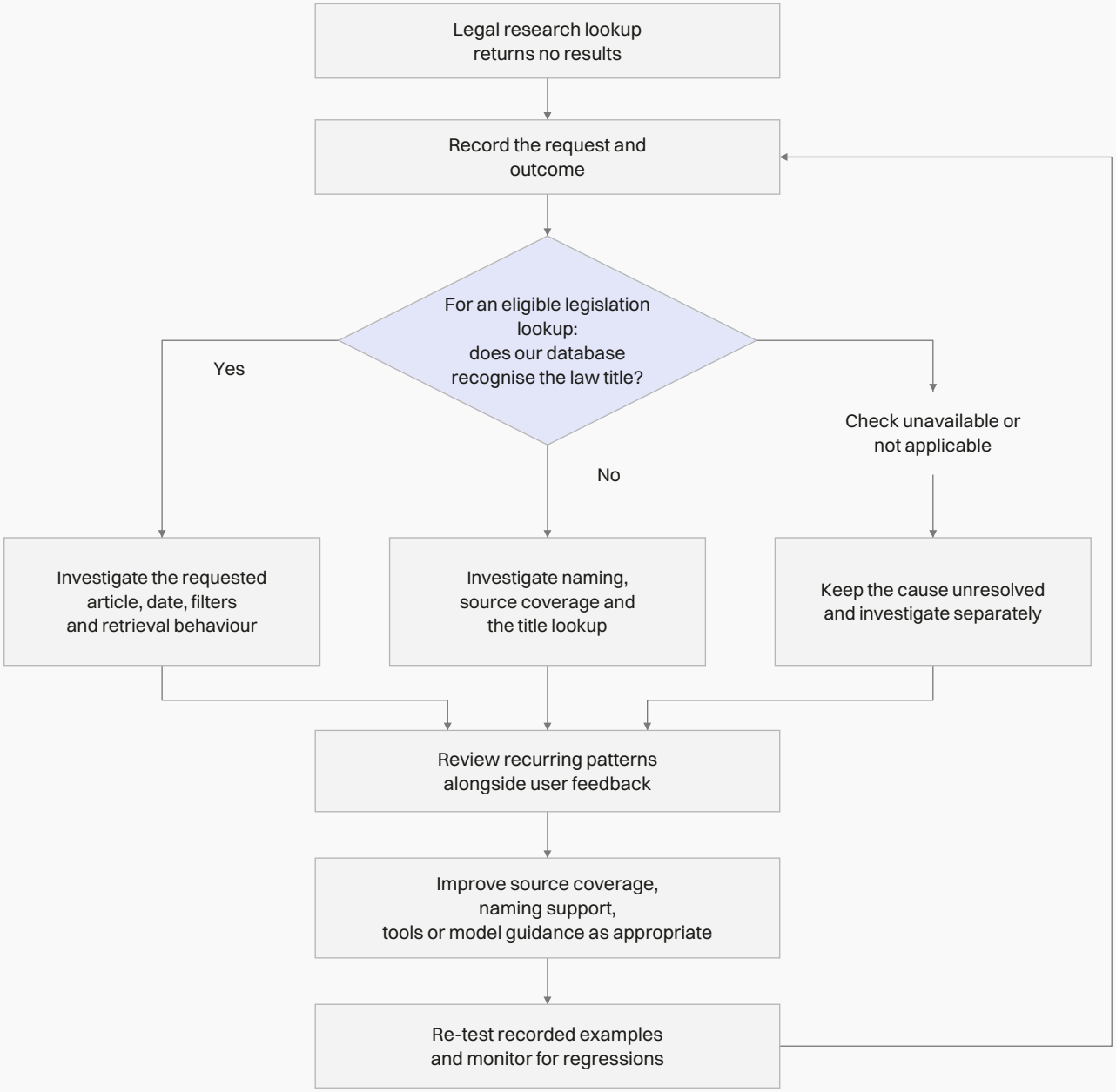


Figure 3. The diagram describes the investigation and improvement cycle we follow, not an automatic hallucination verdict.

OBSERVATIONS FROM THE TOOLS

What the search logs show

This helps us investigate the missing material and wrong references described earlier. Other failures happen even after Zeno retrieves useful evidence, when the model selects, interprets or applies it. Those need their own safeguards.

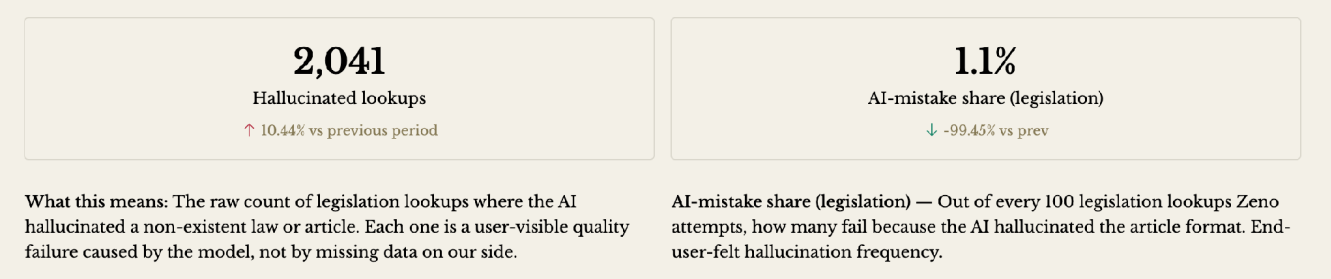


Figure 4. Historical data from the past 3 months in Zeno (July-Sept 2026) illustrating the percentage of “hallucinated” model searches. This is not illustrative of the actual hallucination rate, just the tool described above.

Hallucinated (law, article) patterns — Top law/article combinations the AI repeatedly hallucinates. Counted by request volume, with distinct orgs and runs affected.

Hallucinated (law, article) patterns

	Law	Article	# Requests	# Runs	# Orgs	Last seen
0	Wetboek van Burgerlijke Rechtsvordering	843a	73	71	33	2026-08-25T17:38:45
1	Invorderingswet 1990	36	25	16	11	2026-09-14T14:25:14
2	Invorderingswet 1990	35	20	13	8	2026-09-14T15:41:55
3	Invorderingswet 1990	34	19	14	10	2026-09-14T11:55:12
4	Invorderingswet 1990	25	18	9	6	2026-09-14T22:11:42
5	Algemene Verordening Gegevensbescherming	28	17	17	12	2026-09-14T15:53:05

Figure 5. Historical data from the past 3 months in Zeno (July-Sept 2026) illustrating hallucinated law/article patterns. These are the top Dutch legislation articles which the model could not return results for. Article 843a Rv is shown as the top candidate with 73 requests in the past 3 months.

READING THE FIGURES

The displayed 1.1% concerns the legislation-lookup measure described here; it is not an overall answer-hallucination rate.

FAILURE MODES

Why mistakes can still happen

Going back to the research-folder analogy, several things can go wrong before your colleague writes an answer. They might receive the wrong case, a passage that leaves out an important qualification, or an incomplete account of your situation. They might also have the information they need and still draw the wrong conclusion.

Those differences matter when we design Zeno. Each calls for a different kind of safeguard.

Invented sources and quotations

A language model can produce a plausible-looking legal reference from patterns it learned during training. That does not establish that the source exists, that the wording is accurate, or that the law is still current.

We guide Zeno to retrieve and use legal sources rather than rely on remembered legal knowledge. Connecting a reference to a document gives us something concrete to inspect. Returning original passages also avoids asking the model to recreate the source's wording from memory.

That helps with fabricated references and quotations. It still leaves us with the question of what the source means.

Missing qualifications and relevant material

A real passage can give an incomplete picture. In my situation, a passage about the seller's duty to disclose might have seemed decisive. But material about what the buyer knew, or what should have prompted further investigation, could change the analysis.

This is where passage selection becomes important. We return original text from the selected sections and preserve the connection to the source so it can be inspected in context. Through the knowledge graph, Zeno can also follow recorded connections to related legal material.

These choices help bring relevant evidence into the research. They do not guarantee that the model selects every passage it needs. The model selecting the evidence can make mistakes too.

Wrong context and assumed facts

Even a faithful summary of a judgment can be unhelpful if the circumstances differ from the question being asked. The applicable version of the law may differ. The parties may have exchanged different information. An apparently similar case may turn on a fact the user has not supplied.

We guide the research around the question's facts, jurisdiction and relevant time period. But we also depend on the information available in the conversation. If I leave out that we raised concerns before signing, the model may reason from an incomplete account of what happened.

That is why context belongs in this technical discussion. It affects both the evidence Zeno searches for and how the model applies it.

Drawing a conclusion the evidence does not establish

There is one further step: using the research to answer the question.

A model might find a relevant judgment and describe it accurately, then go too far when applying it. "The buyer succeeded in this case" can become "you should succeed too", without establishing that the same conditions are met.

Software can check certain things directly, such as whether a reference resolves to a document. Assessing whether a legal conclusion follows from that document involves interpretation. Instructions and relevant evidence help guide the model, but neither makes that interpretation infallible.

This is why we work on both the research process and the review of the resulting answer. Better evidence gives the model a stronger basis for reasoning. We still need to examine what it does with that evidence.

THE LAWYER'S CONTEXT

Give the model the context you have

When talking with Zeno, I knew details of my situation that I had not necessarily included in my initial prompts. For example, only I knew that we had raised concerns initially about the foundations before any contract was signed and received verbal reassurance from the seller's agent. A model cannot reliably distinguish an omitted fact from one that is unknown.

Include the circumstances that could affect the analysis. For a question like mine, that might include who made a statement, what they said, when they said it and which reports we had seen. Mark facts as unknown or disputed where appropriate.

A practical addition to your prompt is:

> "If a missing fact could change your conclusion, ask me about it or explain how the answer depends on it. Don't assume it."

The aim is to help the model compare the sources with the situation you are asking about. More text is not automatically better; relevant facts and clear uncertainty are what matter.

HYPOTHETICAL EXAMPLE

What this means for your advice

Suppose an AI answer states that a buyer received no warning about a defect, although the file does not establish that. A real judgment about disclosure duties cannot repair that invented fact. The resulting advice could overstate the buyer's position and influence whether the client pursues the dispute. Correcting it also takes time: the lawyer must revisit the evidence and any conclusions built on the assumption. The citation makes the answer look substantiated, while the underlying problem remains.

PRACTICAL GUIDANCE

Before you rely on an AI answer

Opening the sources helped me prepare for conversations with my lawyer. I could examine the conditions behind a reassuring answer and consider whether the facts of my situation met them.

That is also where the engineering challenge continues. Once Zeno has found the evidence and written an answer, we need to check whether the cited source supports the claim attached to it. In the next article, I'll explain how we approach that check, what its signals mean for a lawyer, and how we test a checker that can make mistakes of its own.

BOX 2 / PRACTICAL CHECKS

1. Separate facts from assumptions.
Check factual statements against the file. Mark anything unknown or disputed instead of allowing it to become an established fact.

2. Open the original sources.
For each material legal proposition, read the supporting passage and its surrounding context. Check whether it records the court's conclusion, a party's argument or a qualification.

3. Check whether the authority applies.
Consider the jurisdiction, the relevant date and version of the law, subsequent treatment and any material differences in the facts.

4. Look for what could change the answer.
Check for exceptions, contrary authority and missing evidence. Asking AI to identify these is useful, but does not establish that its search was complete.

5. Resolve gaps before using the output.
If a material claim cannot be verified, leave it out or investigate further. A confident answer, a working citation or a 'Supported' label is not a substitute for checking it.

The Dutch Bar also recommends manually verifying AI-generated citations, case law and facts before use.[5]

REFERENCES

1. NIST, AI 600-1 (2024), section 2.2: Confabulation.

2. Maynez et al. (2020), On Faithfulness and Factuality in Abstractive Summarization.

3. OpenAI (2025), Why language models hallucinate.

4. Baker and Kanade (2000), Hallucinating Faces. An early use, not proof of who coined today's meaning.

5. Nederlandse orde van advocaten, Aanbevelingen AI in de advocatuur: verify citations, case law and facts manually.



About Samantha Swift

Samantha Swift leads Zeno's product engineering team. Her work includes agentic workflows and scalable machine-learning systems. LinkedIn profile

Engineering Note 01 · September 2026 · More from Zeno