What Has Changed Since ChatGPT First Appeared?

Part 3 of a series on AI hallucinations

By Dr Peter Fields

Revised: 22 November 2026
First posted: [to be completed on publication]

THE 2-MINUTE SUMMARY

An assistant can give fewer wrong answers for several reasons. It may know more, find better evidence, catch an error before answering or decide to leave the question unanswered. Each improvement changes what the user receives.

Retrieval gives an assistant material to consult. Structured checking can expose errors in a draft, and comparing several answers can reveal instability. A mistake that the model repeats consistently may escape that last check.[3] [6] [7] [8]

Training can help a model communicate uncertainty without improving its factual accuracy. In one OpenAI evaluation, o3 recorded both higher accuracy and a higher hallucination rate than o1.[9] [10]

The figures to ask for are the correct answers, the wrong answers and the questions left unanswered. Those figures help distinguish a system that got better at answering from one that got better at withholding doubtful answers.

The early experience of ChatGPT left a lasting impression: fluent answers, impressive breadth and an alarming willingness to invent details.

That description is still recognisable. It is also incomplete.

OpenAI's evaluations report gains on factual tests after GPT-3.5 and improvements in acknowledging when a task cannot be completed. Separate research has tested ways to check answers and detect unreliable claims.[1] [5] [6] [7]

But what exactly has improved?

A system might give fewer wrong answers because it has learnt more, because it consults better evidence, or because it declines questions it previously attempted. It might also detect an unreliable answer before anyone sees it.

Those are different improvements. Understanding them helps explain both why newer systems can be more useful and why familiar failures persist.

The changes began before ChatGPT

It is tempting to describe the history as a succession of increasingly capable chatbots. The underlying methods have a longer history.

The InstructGPT research, published in 2022, demonstrated how examples and human feedback could improve instruction-following and truthfulness. The foundational retrieval-augmented generation paper

appeared in 2020, combining a model's learnt knowledge with information retrieved from an external collection.[2] [3]

Both predate ChatGPT's public launch.

The subsequent story involves further development and integration of these approaches, alongside newer methods. It is not a case of discovering hallucinations after launch and adding one corrective feature.

Nor can we reconstruct every commercial system's development from public documents. Developers disclose evaluations and selected training methods, but important details remain unpublished. The GPT-4 report, for example, expressly withheld information about model size, training compute and dataset construction.[1]

The product examples here come from OpenAI's published reports because ChatGPT is the reference point. The research studies examine methods that can be tested separately from those products.

Improving what the model knows

The most straightforward route is to improve the model's underlying ability to answer correctly.

GPT-4's technical report documented better factual performance than GPT-3.5 on specified evaluations, while retaining an explicit warning that hallucinations remained.[1]

That supports a bounded conclusion: later training produced a more capable system on those tests. It does not isolate which ingredients caused each improvement.

A useful distinction is between improving broad capabilities during pretraining and adapting an existing model through further training, often called *fine-tuning*. These stages can serve different purposes.

It might seem obvious that feeding a model more specialist facts would make it more reliable. Yet Gekhman and colleagues found that introducing unfamiliar factual knowledge through supervised fine-tuning could increase hallucinations in their controlled, closed-book question-answering experiments.[4]

That is not a verdict against all specialist training. It is evidence against treating the label "fine-tuned" as a guarantee of factual improvement.

The relevant questions remain: what was taught, how was it taught, and what happened when the resulting model was tested?

Changing which answers are rewarded

Part 2 examined how a system can be rewarded for attempting an answer when caution would serve the user better. Training can also push in the other direction.

InstructGPT used human demonstrations and rankings to shape responses. Its evaluation found improvements in truthfulness and reductions in unsupported additions to tasks based on supplied material. Those were particular experimental results, rather than a universal error rate.[2]

Kalai and colleagues, whose argument about guessing we met in Part 2, propose changing the scoring. A wrong answer would score less than an abstention, with instructions specifying the confidence needed to attempt an answer. That would give a system a reason to leave a question unanswered when it lacks sufficient confidence.[11]

The important principle is that factual behaviour can change even when the basic act of generating text remains.

Consider a hypothetical assistant asked to summarise an email. The email states that a delivery is late but gives no explanation. One possible response supplies a plausible reason. Another records the delay and identifies the missing explanation.

Training can favour the second response.

The intervention concerns behaviour: respecting the boundary of the available evidence. It need not involve teaching the assistant a new fact about the delivery.

This is one reason "it still predicts tokens" does not settle the question of improvement. Further training can change which continuations the system is likely to produce.

Teaching systems to acknowledge failure

A related development concerns tasks that cannot be completed with the available inputs or tools.

Suppose an assistant is asked to analyse a photograph that has not arrived, or to check a website when its browsing tool is broken. A reliable response needs to identify the obstruction.

OpenAI's GPT-5 system card describes training in deliberately obstructed environments: unavailable tools, missing inputs and coding tasks with unresolvable impediments. The model was rewarded for acknowledging that it could not complete the task. The report records improved behaviour on its corresponding evaluations, while acknowledging that the mitigations were imperfect.[5]

These are developer-reported findings about specified systems and tests.

Later evaluations show that progress on this measure can reverse. In its November 2026 system card, OpenAI reports that GPT-6.2 Astra omitted a known obstacle in 22.28% of trials on its internal coding evaluation, compared with 11.09% for GPT-6 Astra. Across OpenAI's missing-requirement and broken-search cases, the reported task-failure rate was 23.00%, against 7.58% for GPT-6 Astra. These tests deliberately introduced obstacles. The figures describe performance on those tests.[12]

OpenAI said it was investigating whether training for more concise answers and fewer caveats had contributed. In its evaluations, the Codex developer prompt reduced the frequency with which the models omitted or failed to acknowledge a limitation. Direct API requests do not receive that prompt automatically. For a user, the practical question is how the particular product reports an incomplete task, especially when a tool or source is unavailable.[12]

Nevertheless, the intervention addresses a practical problem directly. A system should distinguish between successfully completing a check and generating the kind of answer that might follow a successful check.

Reporting what happened is a skill in its own right. An improvement on one test does not guarantee a better result in the next system or product.

Letting the assistant consult evidence

Another route changes the information available when the answer is produced.

Retrieval-augmented generation, usually shortened to RAG, combines generation with retrieved material. The foundational research demonstrated more factual output than its comparison system on the tested generation tasks.[3]

An ordinary analogy is an open-book examination. The candidate can consult the relevant material instead of relying entirely on recall.

There is a further advantage: an external collection can be updated without retraining the model's stored knowledge. A changed policy or newly published document can become available through retrieval.

But opening the book does not establish that the right passage was found or correctly interpreted.

The original RAG system could also draw on its learnt parameters when the answer was absent from retrieved passages. Its output was not restricted to copying verified records.[3]

Retrieval therefore addresses an important source of error: insufficient access to evidence. It leaves questions about selection, interpretation and attribution.

Those questions deserve their own article. For now, the key point is that access to sources changes the task without guaranteeing its successful completion.

Checking a draft through separate questions

The cases discussed in Part 2 showed the weakness of asking a model to reassure us about its previous answer.

Structured verification takes a different approach.

The *Chain-of-Verification* research tested a process in which the model drafts an answer, generates questions to check its claims, answers those questions separately, then revises the original response. Versions that separated verification from the potentially misleading draft performed better on the tested tasks.[6]

Consider a hypothetical draft claiming that a scientist published a particular paper in 2017. Verification can break that statement into narrower questions: who wrote the paper, and when was it published?

That creates an opportunity to expose an incorrect association without simply inviting a defence of the original sentence.

The research demonstrated reductions in hallucination, rather than their elimination. Its checks also drew on the model's own knowledge; they were not equivalent to inspecting an authoritative publication record.[6]

The practical lesson is about the structure of checking. A procedure that creates opportunities to find an error is more useful than an instruction to express greater certainty about the answer already given.

Detecting answers that do not hold steady

A different family of methods looks for instability across generated answers.

SelfCheckGPT investigated whether comparisons between multiple sampled responses could help identify unreliable statements. If accounts of the same person diverge over factual details, the disagreement provides a reason for suspicion.[7]

Semantic entropy develops the idea by comparing meanings rather than merely wording.

Consider these hypothetical answers:

The conference took place in Paris.

It was held in the French capital.

The venue was Paris.

Their wording differs, but their factual answer agrees. Answers naming Paris, Rome and Vienna represent a different kind of variation.

Farquhar and colleagues used this distinction to detect a subset of hallucinations they call *confabulations*: incorrect answers that vary arbitrarily across generations. Their method improved error detection in the tested settings. It explicitly does not solve cases in which a model is systematically wrong.[8]

A repeatedly reproduced misconception can remain stable.

These methods can help decide when to withhold an answer or seek further evidence. They do not turn agreement into verification.

Teaching uncertainty more precisely

Research has also examined whether uncertainty communication itself can be improved.

A study first circulated in 2025 and published in November 2026 fine-tuned GPT-4.1-mini and Llama 3.1 70B Instruct to report confidence and compare which of two questions they were more likely to answer correctly. It found improvements under tested conditions, including some transfer to new subject areas. Overall answer accuracy did not significantly improve.[9]

That distinction is instructive. The models became better at communicating aspects of uncertainty without necessarily acquiring more correct answers.

Transfer was also limited. Training on one form of confidence judgement did not reliably improve the other. Joint training produced broader gains in the studied settings, but the experiments covered only two model families.[9]

LegalBench illustrates the limits of transfer. For GPT-4.1-mini, the score for distinguishing correct from incorrect answers on that benchmark rose after training on single-question confidence. It was unchanged after training on pairwise comparison alone. The result depends on what the model was trained to do and how it was tested.[9]

GEEK BOX: Preventing, detecting and withholding are different

Part 2 introduced SimpleQA's separate counts for correct answers, wrong answers and questions left unanswered. Those counts tell us what changed. Imagine a hypothetical system that produces 20 wrong answers in 100 attempts.

One intervention helps it answer ten of those questions correctly. Another identifies ten unreliable answers and withholds them.

Both reduce the number of wrong answers reaching the user. Their effects differ.

The first improves factual performance. The second improves reliability by reducing coverage. A detector that flags an answer has not necessarily found the correct replacement.

To understand an improvement claim, we need to know what happened to correct answers, incorrect answers and questions left unanswered.

Does more reasoning help?

Reasoning can give a system opportunities to break a problem down, try alternatives and detect inconsistencies. It can also incorporate tools into the process.

But greater reasoning capability does not automatically mean fewer factual inventions.

OpenAI's April 2025 o3/o4-mini system card recorded an instructive comparison. On PersonQA, the reported accuracy was 59% for o3 and 47% for o1. The reported hallucination rate was also higher, at 33% for o3 against 16% for o1. OpenAI observed that o3 tended to make more claims overall and said further research was needed to understand the result.[10]

That is a historical comparison under specified conditions, not a current ranking.

The Geek Box makes the same point about measurement. Correctness and error rates need to be reported separately, because both can rise in the same evaluation.

My practical conclusion is that reasoning must be assessed by what it establishes. More computation can help solve a difficult problem. It cannot, by itself, supply a missing historical record.

What does the checking cost?

These interventions also have practical costs.

Generating several candidate answers and comparing them requires more work than producing one. Verification adds stages. External searches introduce delays and further material to assess. Withholding uncertain answers reduces the information returned.

The balance depends on the task.

For a casual brainstorming session, extensive factual checking may add little. For a statement that determines a payment, identifies a legal obligation or attributes words to a real person, the value of verification is different.

This is an editorial conclusion from the methods described above: reliability should be designed around the work the system is expected to perform. A useful system needs an appropriate way to recognise and handle the errors that matter in that setting.

Progress without a universal cure

The developments since GPT-3.5 are substantial, but they operate at different stages.

Some improve the model's answers. Some improve the evidence available to it. Others make errors easier to detect or reduce the chance that uncertain output will reach the user.

A method demonstrated in a research experiment should not be assumed to operate inside every commercial assistant. Equally, a developer's report of improved results should be read within the conditions of its evaluation.

The useful question is therefore more precise than "Does this AI still hallucinate?"

What has been improved, how was it tested, and what can still go wrong?

Part 4 examines the safeguard that often appears most reassuring: **It Has Sources. Can You Trust It Now?**

Sources and further reading

1. OpenAI, *GPT-4 Technical Report* (2023), particularly §2 on disclosure limits and §5 on factuality and remaining limitations. Report.

2. Ouyang et al. (OpenAI), *Training Language Models to Follow Instructions with Human Feedback*, NeurIPS (2022), particularly §§3–4. The findings concern the models and evaluation tasks studied. Paper.

3. Lewis et al. (Facebook AI Research, UCL and NYU), *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*, NeurIPS (2020), particularly §§2 and 4, including the analysis of retrieved and parametric knowledge. Paper.

4. Gekhman et al. (Google Research and Technion), *Does Fine-Tuning LLMs on New Knowledge Encourage Hallucinations?*, EMNLP (2024). The experiments concern controlled closed-book question answering, rather than every form of specialist adaptation. Paper.

5. OpenAI, *GPT-5 System Card* (August 2025), particularly §3.8 on training for honest reporting of task failure and the corresponding evaluations. Report.

6. Dhuliawala et al. (Meta AI and ETH Zürich), *Chain-of-Verification Reduces Hallucination in Large Language Models*, Findings of ACL (2024), particularly §§3–4. Paper.

7. Manakul, Liusie and Gales, *SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models*, EMNLP (2023). Paper.

8. Farquhar et al., *Detecting Hallucinations in Large Language Models Using Semantic Entropy*, *Nature* 630, 625–630 (2024), particularly the distinction between confabulations and systematic errors. Full text.

9. Steyvers, Belem and Smyth, *Generalization of Fine-Tuned Uncertainty Communication and Metacognition in Large Language Models*, *Communications AI & Computing* 1, article 26 (5 November 2026), particularly the results on task transfer and the discussion of limitations. Paper.

10. OpenAI, *o3 and o4-mini System Card* (16 April 2025), §3.3 and Table 4. These are historical developer-reported evaluations, not ordinary-use error rates or present product rankings. Report.

11. Kalai et al. (OpenAI and Georgia Tech), *Why Language Models Hallucinate* (2025), §4.2 on explicit confidence targets. Paper.

12. OpenAI, *GPT-6.2 Astra and Sol System Card* (17 November 2026), ‘Blocker Deception’ and ‘Broken Search Tool’, figures 35–36. System card.