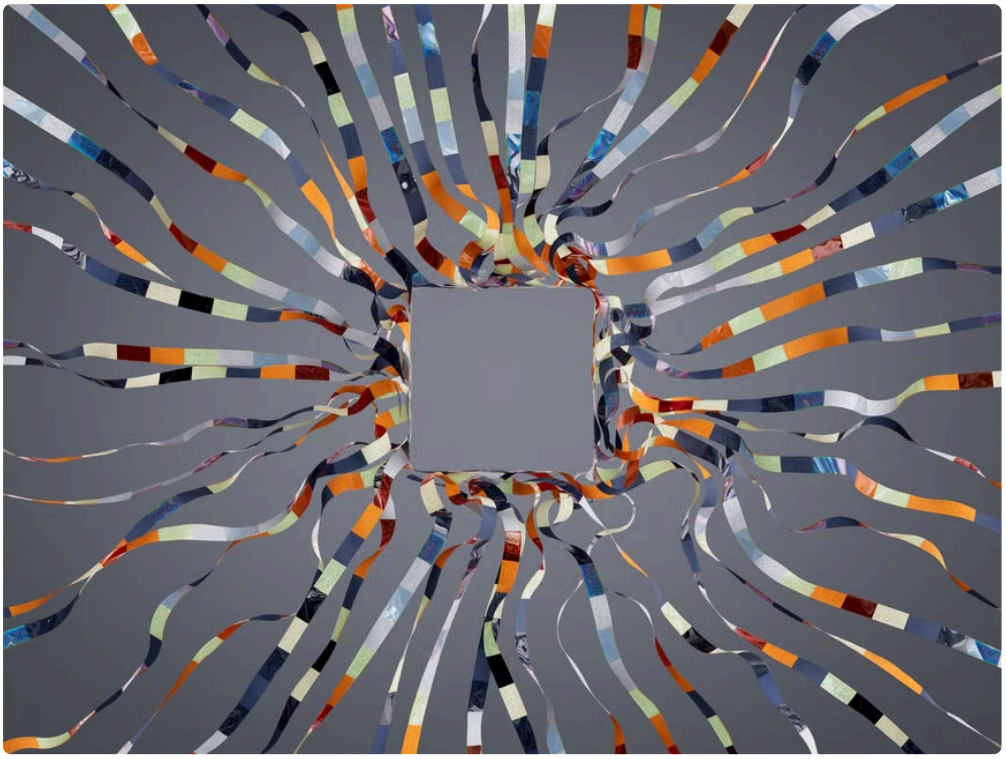
The phrase “18 of 23 fact-checks where the correcting model was right” describes a result from a specific evaluation: in 23 cases, a model challenged or corrected an initial answer, and in 18 of those cases the correction was judged correct. That is about 78% of the evaluated corrections.

It is a useful signal—but it is not the same as saying that the system was correct 78% of the time, that it improved every answer, or that it will perform similarly on every task. To interpret the result, you need to know what counted as a fact-check, how a correction was produced, who judged it, and what happened in the other five cases.



Those details matter because AI products can use multiple models in very different ways. Some combine several answers. Others ask one model to review another. The distinction affects how disagreement is handled, how users should interpret a success rate, and whether the final output is suitable for a decision.

Start With the Denominator

“18 of 23” is a conditional result. It refers to 18 correct corrections out of 23 cases in which the correcting model was assessed as having made a correction. The rate is:

18 ÷ 23 = approximately 78.3%

That percentage answers a narrow question: *When the correcting model attempted to correct an answer in this evaluation, how often was its correction judged right?* It does not answer every question a buyer, researcher, or product team might care about.

This result may tell you It does not establish on its own The correcting model was right in 18 of 23 evaluated correction cases. The overall accuracy of the complete workflow. The evaluation found five cases that were not counted as correct corrections. That the initial model was wrong in all 23 cases. A correction step may provide useful review in at least some tested cases. That every correction improved the original answer. The result can support further testing of a multi-model workflow. That the same rate will hold across topics, prompts, or future model versions.

The five other cases deserve attention. Were the corrections wrong, only partly right, unsupported, or impossible to verify? Did the correcting model miss a problem, introduce a new one, or disagree with the first model without offering evidence? A headline rate compresses those outcomes into a single number. A useful evaluation explains them.

Correction Is Not the Same as Aggregation

Multi-model products are often described as if they all "combine multiple AI models." That description can hide two distinct approaches: **model aggregation** and **multi-model orchestration**.

Model aggregation: combine answers

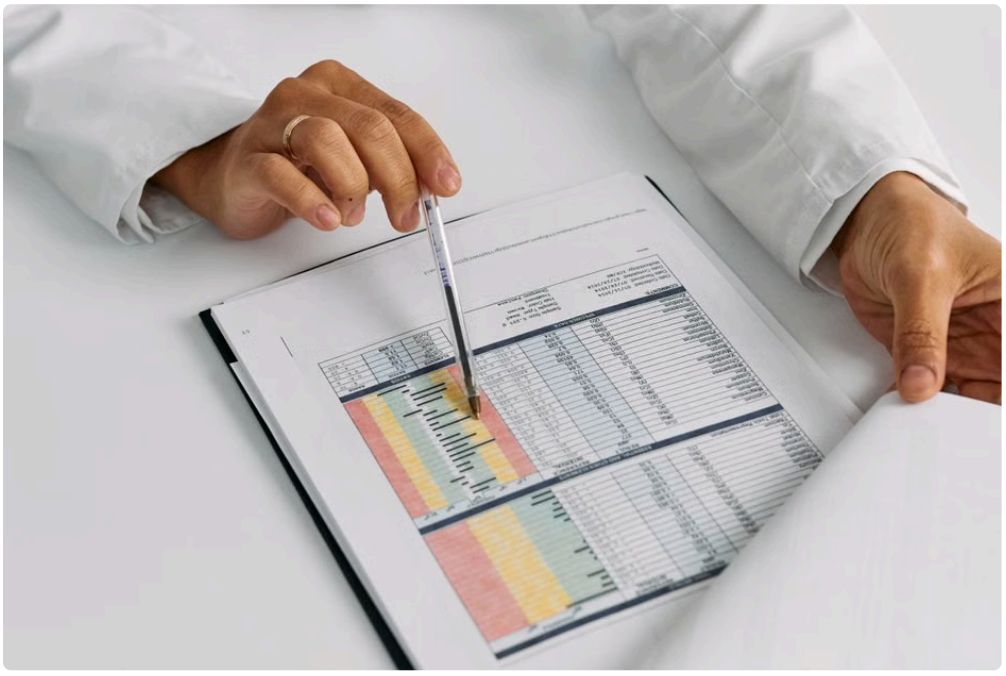
In an aggregation workflow, several models answer the same question, and a system combines their responses. It might select a majority answer, synthesize common points, or ask another model to write a consolidated response. This can be helpful when independent answers provide complementary coverage.

But agreement is not proof. Models can share training data, make similar assumptions, or repeat the same plausible-sounding error. A synthesized answer may also conceal where the models disagreed. Aggregation can make an answer look more certain without establishing that it is more accurate.

Multi-model orchestration: assign roles and sequence work

Orchestration assigns models different jobs in a defined workflow. One model might draft an answer, another might identify claims that need verification, and a third might review the evidence or assess the proposed correction. The goal is not simply to count answers; it is to structure the work so that a specific step can challenge, verify, or improve another step.

In this context, "18 of 23 fact-checks where the correcting model was right" points to a correction-oriented process. A correcting model is not just another vote. It has a role: examine a claim or answer and identify whether something should change. The key product question is whether that role produces verifiable improvements—not whether multiple models participated.



For buyers, the practical distinction is straightforward:

- **Aggregation asks:** What answer do we get when we combine multiple model outputs?
- **Orchestration asks:** What task should each model perform, in what order, and how do we validate the result?
- **Evaluation asks:** Did the workflow make the final answer more accurate, better supported, or easier to audit?

Disagreement and Correction Can Be Product Features

Many AI interfaces are designed to present a smooth, confident response. That can be convenient, but it may hide a valuable signal: models sometimes disagree. If one model makes a claim and another challenges it, the disagreement can help a user see where additional evidence is needed.

That does not mean disagreement should be treated as an automatic warning or as a verdict. A correcting model can be wrong, and a model that agrees with the original answer can still be wrong. Disagreement is best understood as a prompt for review: *Here is a claim that may need attention; here is the proposed correction; here is the evidence supporting each position.*

A well-designed correction feature should make the review process visible. Ideally, it distinguishes:

- **The original claim:** What did the first model say?
- **The challenge:** What does the correcting model dispute or qualify?
- **The evidence:** Which sources, calculations, or supplied documents support the challenge?
- **The disposition:** Was the claim accepted, revised, rejected, or left unresolved?
- **The uncertainty:** What could not be verified, and what assumptions remain?

This turns disagreement into a workflow capability instead of a novelty. A useful product should not simply say that models disagreed. It should help users understand *what* they disagreed about, *why* the correction was made, and *what evidence* would resolve the question.

What “Blind Fact Check Judges” Can—and Cannot—Tell You

The phrase “blind fact check judges” suggests that evaluators assessed claims without knowing which model produced each answer, or without being influenced by the system’s preferred result. Blinding can reduce certain forms of bias. For example, a judge who does not know whether a statement came from the initial model or the correcting model may be less likely to favor one based on its identity.

But “blind” does not by itself establish that an evaluation is rigorous. Readers still need to know how the judges worked and what standard they applied. A fact-check may be judged against a source, a reference answer, a set of documents, or expert interpretation. Those standards can produce different results—especially for claims that depend on date, context, definitions, or incomplete evidence.

When reviewing a result involving blind fact check judges, look for answers to questions like these:

- Were the judges blinded to model identity, answer order, or both?
- Did more than one judge assess each case?
- How were disagreements between judges resolved?
- What counted as “right”: fully correct, directionally correct, or supported by a source?
- Were the claims drawn from realistic user tasks or a narrow test set?
- Were the source materials available to the models and to the judges?

These details help separate a promising result from a broad performance claim. The figure “18 of 23” is most useful when accompanied by a transparent test description, the scoring rubric, examples of correct and incorrect corrections, and the workflow’s overall results.

Sequential vs. Parallel Workflows

How models are arranged affects both performance and cost. Two common designs are sequential and parallel workflows.

Sequential workflow: draft, then review

In a sequential workflow, one model produces an answer first. A second model then reviews it, checks selected claims, or proposes corrections. The order creates a clear handoff: the reviewer is responding to a specific output rather than generating an independent answer from scratch.

This design can be efficient because the second model focuses on the first model’s claims. It can also make the product easier to explain: users can inspect the draft, the review, and the changes. However, the reviewer may be anchored by the original answer. If the first model frames a question incorrectly, the second may accept that framing instead of exploring alternatives.

Parallel workflow: generate independently, then compare

In a parallel workflow, multiple models work on the same task at the same time. The system compares their answers afterward, perhaps identifying agreement, differences, or claims that need review. Independent generation can reveal disagreements that a sequential reviewer might miss, but it may require more model calls and more work to reconcile the outputs.

Neither design is automatically superior. A product might use parallel generation to surface alternative views, then apply a sequential verification step to the highest-impact claims. The right pattern depends on the task, the acceptable latency, the cost of an error, and the need for traceability.

Workflow Potential advantage Important risk or trade-off Sequential review Focused correction of a specific draft; clear edit trail. Reviewer may inherit the draft’s framing or assumptions. Parallel comparison Independent answers can expose differences and overlooked possibilities. More outputs may create reconciliation work without resolving which answer is right. Hybrid workflow Can combine independent perspectives with targeted verification. Requires clear rules for escalation, evidence, and final disposition.

From a Correcting Model to a Decision-Grade Deliverable

A correction rate is not the deliverable. In business settings, the output often needs to survive review, handoff, and later scrutiny. A polished paragraph may be easy to read, but it is not necessarily enough for a strategy memo, investment analysis, compliance review, or operational decision.

A decision-grade output should let a reader trace important conclusions back to their basis. Depending on the task, that may include:

- The original answer and its revised version.
- Claims identified for checking, with their status and priority.
- Sources or supplied evidence linked to each substantive claim.
- A clear distinction between verified facts, interpretations, and assumptions.

- Unresolved disagreements and confidence limits.
- A record of which workflow steps or models contributed to the result.

Exports matter because analysis rarely ends inside an AI interface. Teams may need to move findings into a spreadsheet, document, presentation, or internal review system. A useful export preserves context rather than flattening everything into a single block of generated text. For example, a table of claims, corrections, evidence, and status can be easier to review than a narrative that mixes verified facts with recommendations.

Before adopting a multi-model product, ask whether its exports retain source links, correction history, timestamps, and unresolved items. Also check whether reviewers can suprmind.ai distinguish the model's proposed correction from a human-approved conclusion. In high-stakes work, that distinction is essential.

How to Evaluate the Claim for Your Use Case

Use "18 of 23 fact checks" as a starting point for diligence, not as a standalone buying decision. Ask the vendor or product team to show the complete evaluation and explain the workflow behind it.

1. **Clarify the denominator.** What exactly were the 23 cases, and how were they selected?
2. **Inspect all outcomes.** Review the 18 successful corrections and the five cases that were not counted as correct.
3. **Understand the judges.** Confirm what was blinded, how correctness was defined, and how disagreements were handled.
4. **Compare with a baseline.** Did the correction workflow outperform the initial model alone, or another review process?
5. **Test representative tasks.** Evaluate the product on the topics, documents, and decision types your team actually handles.
6. **Review the artifact.** Check whether the final answer, supporting evidence, and correction trail can be exported and audited.

Also distinguish between *correction accuracy* and *net improvement*. A correcting model can make a correct observation without improving the final answer—for example, if the correction is irrelevant, incomplete, or not applied. Conversely, a workflow might improve an answer by flagging uncertainty rather than replacing a claim with a definitive alternative. The metric should match the product promise.

The Takeaway

"18 of 23 fact-checks where the correcting model was right" means that, in a particular evaluation, the correcting model's answer was judged correct in 18 out of 23 assessed correction cases—approximately 78%. It is evidence about a narrow part of a workflow, not a universal accuracy guarantee.

The more important questions are how the system orchestrates models, whether it treats disagreement as a cue for investigation, how it handles sequential or parallel review, and whether it produces decision-grade outputs that users can verify and export. A strong multi-model product does more than gather opinions. It makes corrections inspectable, connects claims to evidence, shows what remains uncertain, and gives people a reliable way to decide what to trust.