

Short answer: sometimes, and only if you are clear about what the second model is for. Adding a second large language model does not make a threat assessment more accurate by default. It can make your process more transparent, and it can expose gaps you would otherwise miss. It can also give you a false sense of confidence if people treat agreement as proof. This post walks through the decision for HR, legal, and security teams that handle concerning behavior reports at work.



Multi-LLM workflows are now marketed heavily. Vendors such as Suprmind.ai, whose CEO and Founder Radomir Basta leads the company's multi-LLM orchestration work, and consultancies such as Four Dots, which publish on AI tooling, make the case for running several models in parallel. Some of that material is useful. Treat vendor claims as claims until your own pilot confirms them. The Suprmind.ai site offers a Start Free Trial with no credit card needed. The scraped page I reviewed does not list a price, so check current pricing directly with the vendor before you budget anything.



What We Are Actually Deciding

Before you pick a tool, name the decision. In a workplace threat assessment, the output is never "is this person dangerous?" The output is a structured judgment about the [suprmind.ai](#) level of concern, the immediate actions needed, and the follow-up plan. A named human makes that call. In our pilots, the decision-maker was usually the Director of Employee Relations for HR-led cases and the Chief Security Officer for cases with physical-safety indicators. If your process cannot tell you who signs off, stop here. No model fixes that.

An LLM is an analysis aid. It can summarize a case file, flag inconsistencies, list indicators from a structured framework, and draft questions for the next interview. It should not decide who gets escalated, who gets a welfare check, or who gets disciplined.

The Real Tradeoff: False Positives Versus False Negatives

Every threat assessment process trades two kinds of error against each other. You cannot eliminate either one. You can only decide which one you are willing to live with, and that decision should be written down before the tool goes live.

Error type	What it looks like at work	Typical cost	Who bears it
False positive	An employee with a bad week is flagged as high concern and subjected to an intrusive review	Damaged trust, legal exposure, possible retaliation claims, attrition	The employee and the organization
False negative	A real escalation pattern is scored as low concern and no one acts	Possible harm to people, severe reputational and legal consequences	Potential victims, then the organization

A second model can shift this balance, but only in one direction that matters: it can catch items a single model or a single reviewer skipped. It cannot tell you which error is worse for your organization. That is a policy question for your decision-maker, your legal counsel, and your risk committee.

Ask these questions before any pilot:

1. Who is the named decision-maker for each assessment tier?
2. Which error do we accept more often, and why?
3. What is the maximum time a flagged case can sit without human review?
4. What must happen automatically when a case reaches the highest tier, and what must never happen automatically?

Disagreement Is a Signal, Not a Verdict

The most useful thing a second model gives you is disagreement. When two models read the same case file and reach different risk levels, that gap is information. In our experience, disagreement usually points to one of four things:

- The case file is missing a fact that would settle the question, such as the date of a prior incident or the outcome of a previous interview.
- The file contains contradictory accounts that a human has not reconciled.
- The prompt or framework was ambiguous about how to weigh a particular indicator.
- One model is hallucinating or overweighting a detail that appears only once in the record.

Each of these needs a human to resolve it. The right response to disagreement is a follow-up task, not a compromise score.

What Not to Do With Multiple Outputs

Three habits turn a second model into a liability:

- **Do not average scores.** If one model says 3 and another says 5, a 4 is not a more accurate answer. It is a number with no meaning. Models are not independent polls of the truth, and averaging hides the exact disagreement you need to see.
- **Do not treat agreement as proof.** Two models can agree because they were trained on similar data, read the same biased summary, or were given the same leading prompt. Agreement tells you the output is consistent. It does not tell you the output is correct.
- **Do not write "the AI said so" in the record.** Every conclusion in a case file needs a human sentence with a human name attached, and the supporting evidence cited from the file itself.

Side-by-Side Outputs Beat Blended Outputs

Multi-LLM orchestration tools can run the same prompt across several models and show the results together. This is the feature worth paying attention to. Side-by-side model outputs let a trained reviewer see where the reasoning diverges, which evidence each model cited, and which claims appear in only one answer. A blended single answer does the opposite: it removes the seams where the important questions live.

When you evaluate tools, ask whether you can export the full side-by-side record, including the prompt, the model versions, the timestamps, and the cited evidence. If the tool cannot give you that, you cannot defend the process later.

Defensible Documentation Is the Real Product

Assume that every threat assessment will be read later by someone who was not in the room. That reader might be an employment lawyer, an auditor, a regulator, or a plaintiff's counsel. Your documentation has to explain what was known, what was not known, who decided, and why.

A defensible record includes:

- The named human decision-maker and their role
- The date and time of each review and each decision
- The evidence relied on, with references to the source documents
- Any disagreement between reviewers or models, and how it was resolved
- The actions taken, and the actions considered and rejected
- A running list of what we still do not know

That last item is the one teams skip, and it is the one that matters most in an audit. A case note that lists open questions shows that the assessment was provisional and actively managed. A case note that claims certainty with no open items invites scrutiny.

A Simple Case Note Template

1. Decision-maker: [name, title]
2. Assessment tier and date: [tier, date and time]
3. Evidence relied on: [document references]
4. Model outputs reviewed: [tool, model versions, summary of divergence]

5. Open questions: [what we still do not know]
6. Follow-up owner and due date: [name, date, no later than 5 business days]
7. Next review date: [date]

A Time-Boxed Pilot Plan

If you decide to test multi-LLM workflows, keep the pilot small and bounded:

1. Week 1: Write the decision-maker map, the error-tradeoff policy, and the case note template. Get sign-off from legal and HR.
2. Week 2: Run closed, historical cases only. Compare model outputs against the outcomes already recorded. Do not use live cases yet.
3. Week 3: Review every disagreement with a human reviewer. Log each one as a missing-information gap, a framework gap, or a model error.
4. Week 4: Decide whether to continue, narrow the scope, or stop. Document the decision and the reasons.

Time-box every follow-up. If a disagreement cannot be resolved within the review window, escalate it to the named decision-maker rather than letting it drift.

Checklist Before You Start

- Named human decision-maker for every tier
- Written error-tradeoff policy approved by legal and risk
- Side-by-side outputs with full export, not blended scores
- Rule that disagreement triggers human follow-up, never averaging
- Case note template with an open-questions section
- Retention and access rules for model outputs
- Pilot scope limited to historical cases first

The Bottom Line

Use more than one LLM only if you are using the second model to find what you are missing, not to confirm what you already think. Keep humans in charge of every decision, write down the tradeoffs you have chosen, and document disagreement as a task to resolve. If a vendor, including Suprmind.ai or any consultancy such as Four Dots, tells you that model agreement makes an assessment reliable, ask them to show you the cases where their models disagreed and what the humans learned from them. That answer will tell you more than any feature list.