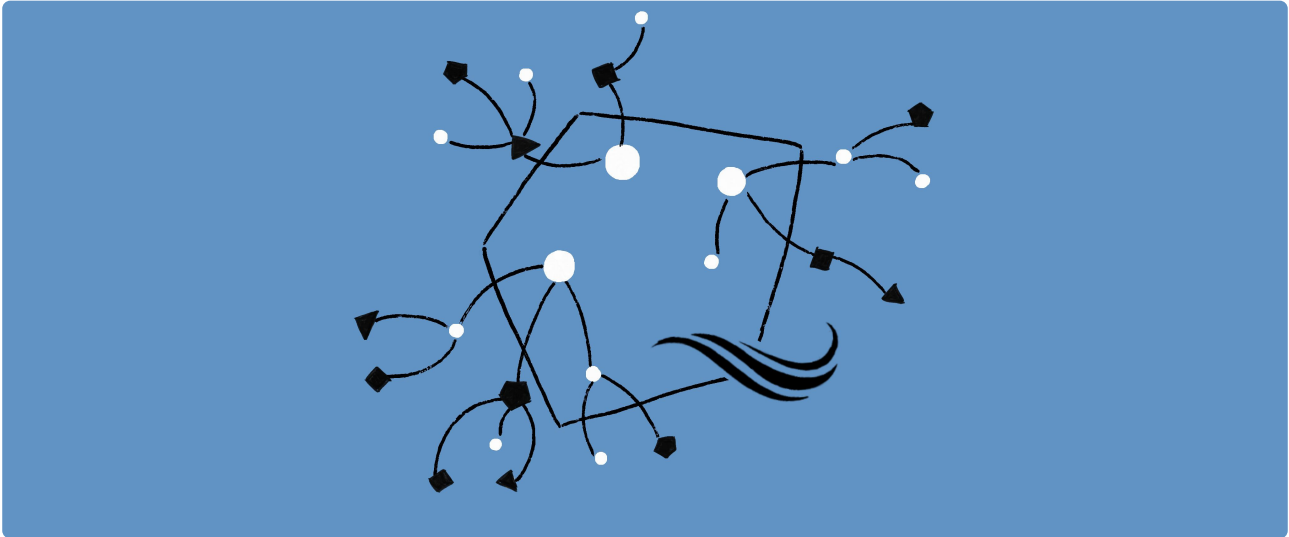


## ESSAY

# Non-determinism is not a bug



In the AI world, non-determinism is a term that gets used almost exclusively negatively. It means that given the same input, a large language model does not produce the same output: something changes, even slightly, every time. The most common response is to try to keep it under control with guardrails, rigid prompts, zeroed-out temperature. And I understand the reasoning, because when you are building a system that needs to be reliable, variability feels like the natural enemy of reliability.

In the first tests we are running with our clients' data, Leo, Da and I tried running the framework in two different modes. In the first, the model received a set of context data and precise instructions on where to look and how to evaluate what it found. In the second, the instructions were almost absent and the model was free to move. The system builds a tree of possible paths, evaluates them with a scoring algorithm that spans the entire universe of available data, and picks the route that leads to the most interesting signal.

Context and instructions help: they provide structure, reduce obvious errors and keep the system focused on what matters. But the free version found things the guided version would never have looked for. Relationships between variables that nobody had connected. Weak signals living at the intersection of data that is not normally looked at together. Anomalies that had no name, and for that reason had never made it into any dashboard.

If our value proposition is to go beyond the perimeter of dashboards, to find what nobody knew to look for, then a purely deterministic system is in some ways an internal contradiction: it always explores the same paths, with the same logic, and always returns the same answer. It is reliable in the same way a dashboard is reliable, meaning it tells you well what you already expected to know, but it is not built to surprise you.

The real question is where control is useful and where it becomes a constraint. The metrics a CFO brings to a board meeting need to be reproducible, and for that job determinism is the right choice. A system that explores data looking for signals nobody has yet thought to look for needs something different: it has to be able to move in ways you cannot fully predict, otherwise it will never do anything a data science team would not have already done. What we are trying to build at Southwind keeps both things separate and makes them work together, because they are two different jobs and mixing them up is probably the most common mistake when building on these models.

I want to be clear about this: I do not know yet how far this idea can be pushed, and the tests we are running are still few. But what we have seen so far has convinced me that treating non-determinism as a problem to solve is the wrong direction, at least for what we are trying to do.