Fee Mustawa: The Right Tier of Intelligence

On the shifting layers of intelligence that interact and make each other.

BY JUDE WHITE

A PICK'n'MIX OF INTELLIGENCE

In 2025 I was doing research in computational neuroscience, where convolutional neural networks[1] had already upended the field, reading patterns in impossible ways and changing our conception of what was possible. No one was much impressed by the flashy, resource-demanding younger brother: large language models.[2] I used them the way most technical people did, for boilerplate code and as a second pass on writing. They could not contribute to anything niche, and were doomed to confuse outdated research and blogs with speculation.

It is with some shock, now, that I spend most working days coordinating not one Claude subagent but a roster approaching twenty. Each is specialised, with multi-turn agency and a growing sense of what I want and how I work. A systems philosopher exists to push me, Socratically, towards thought through designs; a librarian cares about nothing except whether the databases are clean (better for everyone in the office that this isn't a person). A security-minded subagent vets every decision against it's vulnerability to cyberattacks, and a fleet operator keeps several production hubs healthy and talking to each other. Nearly twenty of these, each narrow enough.

Sometimes my job resembles a manager's: making sure my staff have the tools and information to do their jobs well, and stepping in when prudent. Most of the day I review rather than write, several low-grade projects running while I, with my silicon Socrates, work at wherever the conceptual edge is. A single conversation with one model is bottlenecked by how fast one thread can move; a roster working in parallel is bottlenecked instead by how well I define the scope, the success conditions, and the training knowledge of each component, and by my ability to anticipate their habits and their failures. That is what let me build, in months rather than years, this house's actual infrastructure.

The more interesting shift came once that system began to run on its own hardware. The same cloud-hosted models that helped design and review it also 'made Adam from the dust', building and tuning a smaller model (Qwen3 14B) that can run locally and continuously. At that point the agents stopped being only a tool I used to build software and became a component that runs inside it. A subagent that once wrote a parsing function during a development session is a different thing from one that now runs continuously in production,[3] reading incoming reports and resolving names against our ever-expanding entity database. A transient tool, borrowing intelligence from an inconceivable data centre somewhere else, gives birth to essential infrastructure that we run out of the office.

Code is deterministic: same input, same output, every time, because a person wrote down the exact steps in advance. A model is the opposite bet. It has parameters, the internal numbers it tuned itself while training on examples, which it uses to generate responses to new information, a process we call inference. Parameters are what a model learned instead of being told; inference is the model actually doing the guessing. Turn the temperature of a large language model down to zero, though, and it will always emit the most probable next term, so it becomes quasi-deterministic. Confused yet? We can look at the difference between code, inference models and large language models (which are inference models trained on text), through a real production example. We pull live broadcasts from the major Arabic-language stations. Before anything resembling automated political-discourse analysis exists, a chain of narrow tools fires in sequence; some deterministic, most running inference.

First stop is mlx-whisper[4] (large-v3-turbo): audio to text, 809 million parameters, trained on something like five million hours of labeled and machine-labeled speech. The fallibility of these machines, their reliance on their training is revealed here: this model takes an audio spectrograph and predict the word attached to that composition of soundwaves with high accuracy, but if you feed it the breaking-news chime it breaks down and hallucinates the same word 20 times[5]. News chimes, dog barks, and the rest of the audio verse were never in its training set, so it lives in a universe where these cannot be sounds and must instead be strange forms of speech. The options are two. Go Silicon Valley and double down, building a second, super-powerful inference machine trained on all noise to transcribe anything, at the cost of a Saudi Arabian luxury hotel's worth of energy (hyperbole); or simply

write dehallucinate.py: code that knows that Whisper doesn't like chimes and corrects them deterministically. Built by Ranim, our systems intern.

We clearly need a human to measure what these machines infer. But to call Ranim's work of adjudication, building a 'golden set' (the cheat sheet we measure machine performance against), categorically different from what the machines themselves do is generous. It is not outlandish to think of human cognition as an inference machine too[6], running on something like ten to thirty trillion effective parameters that never finish training.[7] That would make the distance a question of architecture and scale, not of kind.

The question of tone analysis gets kicked out of our house to Claude's; to answer something like 'does this presenter seem to implicitly support the actions she is describing' takes more compute than we can reasonably store in our office (at least for now). To do that analysis with a human would demand more labor than we could reasonably ask of anyone, and would drive them nutty besides, however well-informed the duty left them.

Most AI operational mistakes are the same mistake wearing different clothes: confusion about which tier of intelligence a job actually needs. Feed a subagent the working language of a discipline's experts and it can make conceptually unsurprising decisions, because it is reasoning inside that discipline's own vocabulary. A model trained on a narrow slice of audio data still beats any LLM at an audio job, every time. A monolingual English speaker has about as thin a training set for Arabic tone as a model trained to sort fruit by color has for anything else. At the level of language, where actions get described rather than performed, I can trust twenty-plus subagents to work almost unattended, because they are good at language, and language is how we describe what to do. But an LLM is imprisoned in language. Handing a decision to an inference tool with more scale and less relevant data than the job needs is its own kind of mistake. The wheat and chaff of the AI revolution will be those that make Anthropic billions whilst they entrust all operations, technical decisions and knowledge to a supremely self-confident, resource-intensive and non-sovereign large language model, and those who know that Claude and the like offer tools in the rich and varied class of inference models, in the greater taxonomy of variably powerful and intelligent instruments which help turn data into insight.

Notes

1. Convolutional neural networks are a kind of AI model that learns to spot patterns in images and other grid-shaped data by scanning small patches at a time. They powered the last decade's leap in machine vision.

2. Large language models (LLMs) are AI systems trained on enormous amounts of text to predict the next word, which lets them write and answer in fluent language. Claude and ChatGPT are examples.

3. Runtime means the software is running live and doing its real job, rather than being built or tested. A runtime component works on its own, continuously, without someone driving each step.

4. The names that follow (mlx-whisper, pyannote.audio, librosa, wtpsplit, Qwen) are specific, mostly free software tools and AI models, each built for one narrow task. A name ending in .py, like dehallucinate.py, is a small program written in-house in the Python language.

5. In AI, to hallucinate is to output something confident but baseless: words or facts that were never in the input. Here, a speech model fed a non-speech sound invents speech to match it.

6. On the predictive-processing account of cognition (the brain as a hierarchical inference engine minimizing prediction error, not a discontinuous exception to the rest of the roster), see Andy Clark, "Whatever Next? Predictive Brains, Situated Agents, and the Future of Cognitive Science," *Behavioral and Brain Sciences* 36, no. 3 (2013): 181–204, and Karl Friston, "The Free-Energy Principle: A Unified Brain Theory?" *Nature Reviews Neuroscience* 11 (2010): 127–138.

7. On the brain's effective parameter-equivalent scale, see Beren Millidge, "The Scale of the Brain vs. Machine Learning," beren.io, August 6, 2022, https://www.beren.io/2022-08-06-the-scale-of-the-brain-vs-machine-learning/. Millidge estimates 10–30 trillion effective parameters; the more often cited "100 trillion" figure is raw synapse count, not all of which functions as an independently trained weight.

ABOUT CORE GROUP

Core Group is a Beirut-based strategic foresight house. We produce decision-ready analysis and advisory for governments, diplomatic institutions, and strategic investors navigating Middle Eastern complexity. Our work integrates structured analytical products, applied strategic advisory, and analysis-informed mediation; delivered on daily and weekly cycles calibrated to the speed at which the situation changes.

ABOUT PRAXIS

Praxis is Core Group's track on practice and method: the work examined from the inside, where human judgment and machine analysis meet, and what each forecast and each instrument teaches about the method that produced it.

CORE GROUP · BEIRUTCORE.COM · INFO@BEIRUTCORE.COM
ARCHIVE CODE PRX04202607