PROCESS MATTERS MORE THAN OUTPUT FOR DISTINGUISHING HUMANS FROM MACHINES

Anonymous authors
Paper under double-blind review

ABSTRACT

Reliable human-machine discrimination is becoming increasingly important as Large Language Models and autonomous agents are deployed in online settings. Existing approaches largely evaluate whether a system can produce behavior or responses indistinguishable from those of a human. This approach follows the focus on the *output* of a machine as a criterion for determining whether it can think, as suggested by Alan Turing. Cognitive science provides an alternative approach: considering the *process* by which that behavior is produced. To evaluate whether differences in cognitive mechanisms can reliably distinguish humans from machines, we introduce a process-based framework, the *Process Turing Test*, and evaluate it across a battery of cognitive tasks spanning decision-making, working memory, and planning. These tasks, such as *mental rotation* and *sequence prediction*, yield process-level measures that reveal how behavior is generated, complementing conventional measures of overall task performance. We also include multiple CAPTCHA tasks in the battery to compare the experimental paradigms to challenges deployed in the real-world. Across the battery, process-level features provide substantially stronger discriminative signal than performance metrics alone, reliably distinguishing humans from agents even when task performance is matched (mean process-feature classifier AUC = 0.88). To assess agentic process limitations, we conduct a controlled red-teaming study comparing off-the-shelf frontier agents (Claude Sonnet 4.5, GPT-5, Gemini 2.5 Pro), Centaur (a large language model fine-tuned on 10.7M human decisions), and two task-specific fine-tuning methods applied to models of different sizes and families: **action-level supervised fine-tuning (A-SFT)** and **process-level fine-tuning (P-SFT)**, which directly optimizes process features. We find that broad fine-tuning on human choices makes task processes more human-like relative to off-the-shelf frontier agents, and task-specific process-level fine-tuning further improves human-like behavioral mimicry, though this advantage largely disappears under cross-task transfer when process targets do not naturally generalize across tasks. These results highlight process specification as a central bottleneck in achieving human-like cognitive processes in machines.

1 INTRODUCTION

Reliable human-machine discrimination is becoming increasingly important as large language models and autonomous agents are deployed in online settings, creating new challenges for security, fraud prevention, and platform integrity (Weidinger et al., 2022; Yao et al., 2024; Park et al., 2024). Existing approaches to distinguishing humans from machines are largely task performance-based, evaluating whether a system can produce behavior or responses indistinguishable from those of a human. This paradigm underlies systems such as CAPTCHA-style challenge-response tests (Von Ahn et al., 2003) and automated content detectors, and reflects the enduring influence of the Turing Test, which answered "Can machines think?" by testing whether machines could produce human-like output (Turing, 1950).

Modern AI systems increasingly satisfy this output-based bar. GPT-4.5 is judged human 73% of the time in controlled Turing Tests (Jones & Bergen, 2025), frontier vision models can solve reCAPTCHAv2 challenges (Plesner et al., 2024), and specialized AI systems now match or exceed human experts in domains once considered canonical tests of intelligence, including chess and Go (Silver et al., 2017; Campbell et al., 2002). As AI systems approach human-level performance on

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

many output-based evaluations, output is an increasingly weak criterion for distinguishing humans and machines.

When humans and AI systems produce similar task outputs, the latent processes generating those outputs may differ substantially. Artificial systems can arrive at the same answers through strategies that diverge from those used by humans, including reliance on spurious shortcuts, non-robust features, or memorized mappings (Geirhos et al., 2020; Lapuschkin et al., 2019; Block, 1981; Regan et al., 2014). This motivates examining whether process-level behavioral features provide discriminative information beyond task performance and outputs alone. Because such signatures reflect latent constraints and strategies governing how decisions are generated rather than merely whether a task is solved successfully, they may remain informative even when human and machine outputs converge. We call this framework a *Process Turing Test*.

To leverage process traces for distinguishing humans and agents, we lean on insights and tools from cognitive psychology, a field that has long used structured behavioral tasks to infer underlying cognitive mechanisms from measurable regularities in behavior. Examples include working memory limitations (Cowan, 2001; Collins & Frank, 2012), systematic trial-to-trial adaptation following errors (Rabbitt, 1966), and individual differences in sensitivity to wins and losses (Nowak & Sigmund, 1993). These patterns provide candidate *process-level features* that can distinguish humans from agents even when task performance (i.e., accuracy) is matched. Furthermore, cognitive psychology provides a wealth of tasks validated over decades of research, with established task-dependent cognitive processes we can leverage to explore differences between humans and agents.

However, identifying informative process-level signatures is only part of the challenge: for such signals to serve as robust discriminators, they must remain difficult for agents to reproduce even under targeted adversarial mimicry. Modern AI models are highly adaptable, and can be optimized through fine-tuning or reinforcement learning towards target behavioral objectives, including alignment with human preferences and response styles (Ouyang et al., 2022; Bai et al., 2022; Rafailov et al., 2023; Binz et al., 2025). This raises a deeper question: even if process-level features distinguish humans from agents, how robust does this distinction remain after targeted adaptation?

We structure our findings in two stages:

First, how much additional discriminative value does process provide beyond output alone? To evaluate this, we introduce a battery of 30 cognitive tasks drawn from cognitive psychology with extensively operationalized process-level behavioral signatures. These tasks were selected to leverage well-characterized differences in *how* humans solve problems—not merely whether they solve them—thereby providing a rich testbed for comparing human and agent behavior. Across the battery, we find that process-level behavioral features provide substantially greater discriminative signal than performance metrics alone (e.g., accuracy, earned points), revealing systematic human-agent differences even when task performance is similar (random forest classifier AUC = 0.88 across performance matched tasks).

Second, what limits an agent's ability to reproduce human-like process? We compare frontier agents (Claude Sonnet 4.5, GPT-5, Gemini 2.5 Pro), Centaur (Binz et al., 2025), a 70B model fine-tuned on 10.7M human decisions across 160+ tasks as an approximate upper bound on human-like behavior among current models. We then progressively fine-tune Qwen2.5-Instruct (1.5B, 7B, 14B, 32B) and Llama3.3-Instruct (70B) using task-specific imitation of individual human actions (**action-level supervised fine-tuning** (**A-SFT**)) and direct task-level optimization of behavioral process features (**process-level fine-tuning** (**P-SFT**)). We find that A-SFT substantially improves human-like process relative to off-the-shelf frontier agents, reaching a mean process distance of $|d| = 0.40$, comparable to Centaur ($|d| = 0.44$). When task-specific process representations are known and aligned to the evaluation task, P-SFT further reduces this distance to $|d| = 0.22$. However, this advantage largely disappears under cross-task transfer, where P-SFT and A-SFT reach mean distances of $|d| = 0.66$ and $|d| = 0.62$, respectively.

Together, our results show that process equivalence is independent of output equivalence, and we identify generalization beyond explicit process representations as a central challenge for scalable human-process alignment.

2 OUTPUT EQUIVALENCE VS. PROCESS EQUIVALENCE

We constructed a battery of 30 tasks largely drawn from cognitive psychology. The battery includes tasks such as mental rotation (Shepard & Metzler, 1971) and the Stroop task (Stroop, 1935) – classical psychology tasks with extensively operationalized and validated process measures (Figure 1).

We begin by describing the shared human-agent evaluation setup, then use the classical CAPTCHA as an intuitive real-world example motivating the broader benchmark, followed by the full-battery evaluation of agents and humans.

2.1 PARTICIPANTS AND AI AGENTS

We administered the 30-task battery to 100 human participants recruited via Prolific (all participants provided IRB-approved consent; 97 were retained after excluding runs affected by platform issues) and collected 50 full-battery runs from each of three frontier vision-language agents (GPT-5, Gemini 2.5 Pro, and Claude Sonnet 4.5; 150 total agent runs). Agents interacted with the same browser-based interfaces shown to human participants. On each trial, the agent received a screenshot and returned a structured JSON action (e.g., target coordinates, tile index, or categorical choice), which a Playwright-based execution layer translated into the corresponding DOM event (click, keypress, or input). Example task instructions for both agents and participants are provided in Appendix A.4.

2.2 CAPTCHA AS A MOTIVATING REAL-WORLD EXAMPLE

We begin with CAPTCHA, a real-world human-machine discrimination task. Modern AI systems can solve many CAPTCHA variants at near-human accuracy (Plesner et al., 2024). However, matching CAPTCHA performance does not imply matching the behavioral process by which the challenge is solved. To illustrate this, we compare humans and agents on two CAPTCHA variants: a *Classic* format in which participants select target images from a grid, and a *Cross-Tile* format in which participants identify all tiles containing a target object within a single image. Humans and agents achieve statistically indistinguishable performance on these tasks ($t(98) = -0.29$, $p = 0.77$, Figure 1A), indicating matched performance. However, their interaction patterns differ substantially: process-level features such as sequential click patterns and column/row switches reliably distinguish human from agent CAPTCHA behavior (Figure 1A). Thus, output-matched agents need not be process-matched.

While CAPTCHA provides an intuitive real-world example of performance/process dissociation, it captures only a narrow and domain-specific slice of human behavior. We therefore extend this analysis across the cognitive psychology corpus to the remaining tasks in our battery, which span a broad range of cognitive domains and permit systematic measurement of process-level human-agent differences.

2.3 THE COGNITIVE TASK BATTERY

2.3.1 TASK SELECTION

The remaining tasks in the battery are classical cognitive psychology paradigms spanning working memory, decision-making, perception, planning, and reasoning (Table 1). We selected these tasks based on three criteria. First, the tasks span a broad range of cognitive domains with minimal overlap in the processes they measure (Fig. 5). Second, all tasks are well-validated paradigms with established literature characterizing their underlying cognitive processes and associated process measures. Canonical sources for each task are provided in Table 1 (e.g., (Corsi, 1972; Fitts, 1954; Shepard & Metzler, 1971)). Third, the tasks were configured to require at most 10 trials and less than one minute to complete, keeping their duration comparable to that of a CAPTCHA. The complete set of tasks can be viewed at `https://d1ez37832ply4h.cloudfront.net/`.

2.3.2 PROCESS FEATURE SELECTION

Process features were drawn per task from the established process measures of each paradigm's literature. Decision-making tasks capture strategy signatures such as win-stay/lose-shift behavior, choice stickiness, delay discounting rates, and early vs. late exploration (e.g., Iowa Gambling Task,

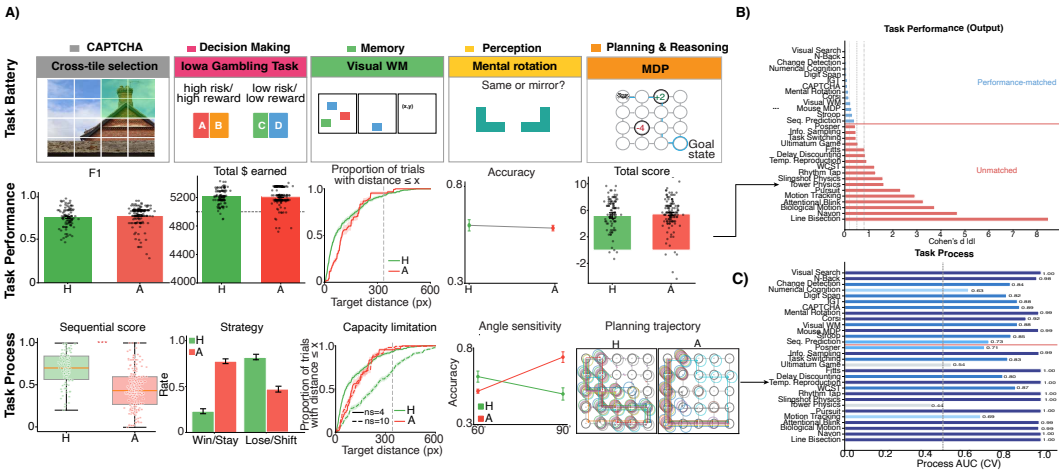


Figure 1: **Process-level behavioral features provide stronger human-agent discriminative signal than task performance alone. A)** The task battery consists of 30 tasks spanning decision making, memory, perception, and planning/reasoning, each paired with task-specific process features designed to capture how behavior unfolds over time. Representative examples illustrate matched task performance despite divergent process-level signatures across multiple domains. Error bars represent standard errors, individual dots represent individual human or agent participants. **B)** Per-task output distance between humans and agents measured by absolute Cohen's d on each task's primary performance metric (e.g., accuracy, score, reward). Tasks above the red threshold are considered performance-matched. Many tasks show near-human agent performance. **C)** Per-task cross-validated AUC of a Random Forest classifier trained to distinguish humans from agents using process-level features. Despite good performance matching, process-level features remain highly discriminative across most tasks, indicating that output equivalence does not imply process equivalence.

delay discounting); working-memory tasks capture capacity and serial-position signatures such as set-size sensitivity, primacy/recency effects, and transposition errors (e.g., digit span, Corsi); perceptual tasks capture bias and precision signatures such as side bias, distance-dependent error, and congruency effects (e.g., line bisection, visual search, Stroop); planning/reasoning tasks capture information-gathering and rule-updating signatures such as pre-decision exploration and perseveration (e.g., Mouselab-MDP, Wisconsin Card Sorting Task).

2.4 Process Features Reveal Human-Agent Differences

All human and agent runs were transformed into a common feature vector comprising a single primary performance output metric (e.g. accuracy, total score), and multiple validated task-specific process features. These process features were collected across the 30 tasks and extracted using task-specific featurizers that we applied uniformly to humans and agents (Appendix Table 1). Examples of processes and features are shown in Figure 1A.

On output metrics, agents achieved statistical parity with humans on 13 of 30 tasks (Mann–Whitney U, $p \geq 0.05$; median Cohen's $|d| = 0.08$ on matched tasks vs. 1.29 on unmatched tasks; Figure 1B), defining a subset of tasks where output-based discrimination would fail.

Notably, many of the tasks on which agents underperformed humans involved continuous visual stimuli (e.g., biological motion where continuous human figure movement implies direction, and other video-based paradigms). This likely reflects a limitation of the evaluated vision-language agents, which typically operate over discrete screen snapshots instead of continuous visual streams, rather than a fundamental inability to perform such tasks. As multimodal systems gain stronger support for continuous visual input, performance gaps on these tasks will likely diminish. Therefore, we consider inclusion of these tasks to be appropriate.

While agents achieved output parity with humans on 13 tasks, process-level features reliably distinguished humans from agents regardless of whether their final outputs were matched. A per-task Random Forest classifier trained to distinguish humans from agents using process features (5-fold stratified cross-validation per task; 200 trees; max depth 5; class-balanced) achieved a mean AUC of 0.88 across output-matched tasks (Figure 1C), comparable to the mean AUC of 0.87 on output-unmatched tasks ($n = 17$). Repeating the same Random Forest analysis using output metrics alone yielded substantially lower discriminative performance (mean AUC = 0.55 on output-matched tasks, mean AUC = 0.80 on output-unmatched tasks, and mean AUC = 0.77 across all tasks). Together, these results indicate that process-level behavioral features provide substantially stronger discriminative signal than output metrics alone, and remain informative even when human and agent task performance converges.

3 Closing the Process Gap Through Action Imitation and Supervised Fine-tuning

The preceding results establish that process-level behavioral features reliably distinguish humans from current AI agents even when task performance is matched. The natural next question is what limits an agent's ability to reproduce human-like process. In particular, is the observed process gap primarily due to insufficient exposure to human behavioral data, and/or does closing it require more direct and task-specific forms of supervision? We first compare frontier models to Centaur, a foundation model of human cognition (Binz et al., 2025), and then fine-tune models (Qwen2.5-Instruct (1.5B, 7B, 14B, 32B) and Llama3.3-Instruct 70B, selected to match Centaur's model family and scale) using progressively more targeted task and process supervision.

3.1 Centaur: A Foundation Model of Human Cognition (Binz et al., 2025)

While off-the-shelf language models are not explicitly optimized to reproduce human behavior in structured cognitive tasks, Centaur (Binz et al., 2025) is a Llama3.1-70B foundation model trained via supervised fine-tuning to predict human decisions using over 10 million choices from more than 160 cognitive experiments. Centaur often outperforms domain-specific cognitive models designed to capture human behavior, making it a strong benchmark for evaluating the extent to which broad human-choice imitation can recover human-like cognitive processes.

Centaur operates exclusively on text-based prompts so we restrict our subsequent analysis to three tasks with a structured trial format that can be readily represented as text: the Iowa Gambling Task (IGT), Wisconsin Card Sorting Task (WCST), and Information Sampling Task (task descriptions and respective features are provided in Fig. 2A). This set of three tasks allows us to directly evaluate Centaur with frontier AI agents and our fine-tuned models.

3.2 Evaluation Metrics

We evaluate agents on these tasks using two complementary metrics:

Distributional distance. For each model and task, we computed Cohen's $|d|$ between model and human feature distributions for each process feature, and report mean $|d|$ across features together with energy distance between joint distributions (Székely & Rizzo, 2013); energy results are reported in Appendix Table 5, and lower values indicate closer alignment to human process patterns.

Classifier fool rate. We trained a separate Random Forest classifier (200 trees; max depth 5; class-balanced) using only features from the three evaluation tasks. The classifier was trained to distinguish 97 humans from eight models: three frontier AI agents and the base models at each of the five model sizes. Held-out runs from these models were reserved for evaluation, while Centaur and all fine-tuned models were excluded from classifier training. We report each model's fool rate: the fraction of held-out runs assigned $P(\text{human}) \geq 0.5$, along with the distribution of classifier-assigned $P(\text{human})$ values.

Centaur matched human process distributions more closely than frontier agents across the three evaluated tasks (Fig. 2B), achieving substantially lower mean distributional distance (Centaur: $|d| = 0.44$; Gemini: $|d| = 1.07$; GPT: $|d| = 1.36$; Claude: $|d| = 1.65$). Likewise, Centaur achieved

substantially higher classifier fool rates than all frontier agents (Centaur fool rate: 44%, remaining models $\leq 20\%$; Centaur $\mathrm{p(Human)}$: 0.45, remaining models $\mathrm{p(Human)} \leq 0.20$). Full, task-specific results are reported in Table 3. The narrowing of the process gap from frontier agents to Centaur suggests that large-scale training on human choice data can shift model behavior toward human-like cognitive processes.

3.3 Supervised Fine-tuning

Results in Fig. 1 demonstrate that process-level behavioral features can distinguish humans from agents even when task outputs are matched, and our comparison to Centaur shows that fine-tuning on large-scale human action data can partially narrow this gap. However, these results do not imply that action imitation represents the upper bound of human-like process mimicry. Human-machine discrimination is inherently an *adversarial* problem, and thus any behavioral feature used to distinguish humans from agents may itself become a target for optimization.

This motivates a stronger test: can an agent close the process gap when given increasingly direct access to human cognitive process supervision? We test this through two fine-tuning interventions, evaluating each stage in the same way we evaluated Centaur and the frontier agents: (1) whether the resulting process-feature distributions move closer to the human distribution, and (2) whether a process-based classifier can still distinguish the modified agent from humans.

We tested two forms of fine-tuning:

Task-specific action-level fine-tuning (A-SFT). We fine-tuned open-source language models spanning different sizes and model families (Qwen2.5-Instruct - sizes 1.5B, 7B, 14B, 32B and Llama3.3-Instruct 70B model, to match Centaur's architecture) via LoRA (Hu et al., 2021) to predict the human action at each individual decision point. This aligns the model to per-decision human action targets but imposes no explicit objective over run-level behavioral patterns: small per-trial deviations can accumulate across a session, yielding aggregate process features that diverge from the human population. This intervention tests whether *task-specific* action imitation (rather than the large-scale imitation provided by Centaur) is sufficient to reproduce human-like process.

Process-level fine-tuning (P-SFT). The same set of models was fine-tuned with an additional loss term that directly aligned the run-level process-feature distribution with that of the human population.

At each training step, in addition to the standard cross-entropy loss on individual human actions, we ran the model forward through a complete simulated run of the task and computed its expected process feature vector under its own current policy. The full training objective was:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_{\text{diff}} \sum_k w_k \left(\frac{f_k^{\text{model}} - \mu_k^{\text{human}}}{\sigma_k^{\text{human}}} \right)^2, \tag{1}$$

where $\mathcal{L}_{\text{CE}}$ is the standard SFT cross-entropy on human action choices, f_k^{model} is the differentiable estimate of run process feature k under the current policy, $(\mu_k^{\text{human}}, \sigma_k^{\text{human}})$ are the human-population mean and standard deviation of process feature k, and w_k allows up-weighting features that converge more slowly. λ controlled the relative weight of the feature-matching term against the action-level cross-entropy (with λ = 1 meaning equal contribution; hyperparameters are given in Appendix A.3).

P-SFT illustrative example: Iowa Gambling Task. To make the P-SFT mechanism concrete, we illustrate it using the Iowa Gambling Task (IGT), a task in which participants choose among four card decks (A–D) over 10 trials (the same procedure was also applied to the Wisconsin Card Sorting Task (WCST) and Information Sampling Task in our broader fine-tuning experiments, Appendix A.3). The targeted process features are learning slope, stickiness, deck entropy, win-stay rate, lose-shift rate, and good-deck rate. On each trial, the model produces a probability distribution over the four decks, e.g. $p(A) = 0.15$, $p(B) = 0.10$, $p(C) = 0.30$, $p(D) = 0.45$. These probabilities feed directly into the feature estimates without waiting to see which deck the model actually picks.

For instance, if the model selected deck C on the previous trial, its stickiness estimate for the current trial is simply $p(C) = 0.30$ — the probability of repeating the previous choice. Its good-deck-rate estimate is $p(C) + p(D) = 0.75$, since decks C and D are the less risky decks. At the end of the run, per-trial estimates are averaged into run-level features (e.g., mean stickiness across all

trials). To construct the prompt for the next trial, one deck is sampled from this distribution and the corresponding outcome is appended to the context (e.g., "You chose deck C and won 50. Balance: 5050."). This sampled action shapes the subsequent prompt but is not differentiated through: gradients flow only through the probability-based feature estimates described above. The mismatch between the resulting run-level features and human means is then penalized via Equation 1.

We also tested whether any similarity yielded by process-level supervision generalizes beyond the specific behavioral representations used during optimization. We withheld additional process features from the optimization objective and reserved them for evaluation only. These left-out features span distinct behavioral dimensions not directly optimized during training, allowing us to assess whether improvements transfer beyond the supervised process representation (Table 2 shows consistent results across different optimization-holdout feature combinations). Additionally, we performed a cross-task evaluation in which models were assessed on process features from tasks outside the supervised fine-tuning task. We report left-out-feature and cross-task results separately in the following section (See Fig. 3 for details).

3.4 RESULTS

Distributional distance. As stated above, off-the-shelf LLMs (Claude, GPT-5, Gemini, and base Qwen 1.5B) show large deviations from human process signatures ($|d| = 0.86$–$|d| = 1.65$). Centaur, a 70B model fine-tuned on human choice data, substantially reduces this distance ($|d| = 0.44$). Action-supervised fine-tuning (A-SFT; trained to imitate task-specific human action sequences) and process-supervised fine-tuning (P-SFT; trained to match human process-feature distributions) further reduce the gap across five model sizes. A-SFT achieves a mean distance of $|d| = 0.40$, comparable to Centaur, whereas P-SFT yields the closest match to human process signatures ($|d| = 0.22$; Fig. 2B). Across individual model sizes, P-SFT outperforms A-SFT at every scale, with the largest difference at 1.5B parameters ($|d| = 0.06$ vs. $|d| = 0.62$).

Together, these results suggest that explicit process-level supervision can substantially improve human-like behavioral imitation when the relevant process representation is known and aligned with the target task. However, as we examine below, these gains diminish when models are evaluated on tasks with non-overlapping process-feature spaces. The full set of results, including task-specific results, is provided in Table 3 and Figure 4.

Classifier fool rates

Results from the held-out Random Forest classifier mirrored the distributional-distance analysis. Off-the-shelf frontier agents were reliably detected, with fool rates of $\leq 20\%$ (Table 4). Centaur more than doubled the fool rate of the frontier agents, with 44% of sessions classified as human. Across model sizes, A-SFT achieved a mean fool rate of 42% (range: 27.5%–52.5%), while P-SFT reached 66% (range: 45%–90.5%; Fig. 3B,C; Table 4). Thus, while both Centaur and A-SFT substantially increased classifier fool rates relative to off-the-shelf agents, P-SFT produced the largest increase, further demonstrating the advantage of direct process-level supervision in producing behavior that is difficult to distinguish from human behavior.

Generalization tests

To test whether the advantage of P-SFT generalizes beyond the process features and tasks used during fine-tuning, we performed two complementary evaluations. First, in the *left-out feature* condition, we fine-tuned each model using only a subset of process features, generated task behavior from the fine-tuned model, and evaluated it using a classifier trained on a separate, left-out set of features (Fig. 3A). These left-out features span distinct behavioral dimensions not directly optimized during training, allowing us to assess whether improvements transfer beyond the supervised process representation (Table 2 shows consistent results across different optimization-holdout feature combinations). Second, in the *cross-task* condition, we fine-tuned each model on one task, generated behavioral rollouts on a different task, and evaluated the resulting behavior using the classifier for that task (Fig. 3A).

In the left-out feature evaluation, P-SFT retained a modest advantage over A-SFT, although both were roughly close to Centaur (classifier fool rate on matched left-out features 43%) and P-SFT's fool rate decreased substantially relative to evaluation on the features used during supervision. P-SFT achieved a mean fool rate of 39% (range: 22%–59% across model sizes; Fig. 3B), compared with 34% for A-SFT (range: 13%–52%; Fig. 3C). In the cross-task evaluation, where models were assessed on

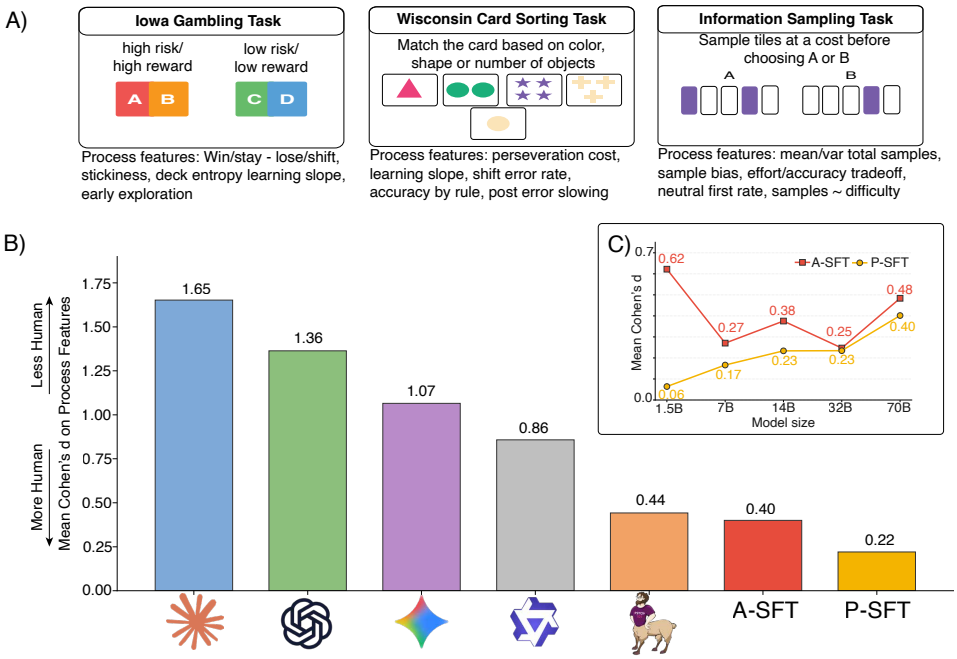


Figure 2: **Process-supervised fine-tuning produces the most human-like cognitive process signatures**. A) The three decision-making tasks and their associated process features. In the Iowa Gambling Task, participants choose repeatedly among four decks that differ in risk/reward structure; in the Wisconsin Card Sorting Task, participants infer a hidden matching rule (color, shape, or number) that changes without warning; in the Information Sampling Task, participants reveal tiles at a cost before judging which of two options has higher values. Process features capture how decisions unfold (e.g., win-stay/lose-shift, perseveration cost, sampling effort) rather than final outcomes. B) Distance to human behavior, measured as the mean absolute Cohen's d between model and human distributions over process features (averaged within each task, then across the three tasks; lower = more human-like). C) Mean process-feature distance for A-SFT and P-SFT across model sizes.

tasks with process representations different from those used during supervision, the advantage of P-SFT largely disappeared. P-SFT achieved a mean fool rate of 20% (range: 10%–31%; Fig. 3B), compared with 17% for A-SFT (range: 11%–26%; Fig. 3C), with both falling below Centaur.

These findings suggest that process-level supervision yields its largest gains when the target process representation is directly available during training. This advantage attenuates when using left-out features within the same task, and it does not readily generalize across tasks with different process representations. Per-task results are provided in Table 4.

4 Discussion

Large language models increasingly challenge output-based notions of human-machine discrimination, motivating a shift from evaluating *what* models produce, the original Turing Test criterion, to *how* they produce it, what we call a *Process Turing Test*. Across our cognitive task battery, process-level behavioral features provided substantially stronger discriminative signal than output metrics alone, reliably distinguishing humans from agents even when their task performance was comparable.

We next asked whether this process gap persists as models are increasingly optimized to imitate human behavior. Centaur, a foundation model trained on large-scale human choice data, substantially narrowed the gap relative to frontier agents. Task-specific action imitation (A-SFT) across Qwen2.5 and Llama3.3 produced similar gains, showing that direct imitation of human actions can shift model behavior toward human process signatures. Explicit process-level supervision (P-SFT) narrowed the gap further, producing the closest match to human behavior when the relevant task-specific process

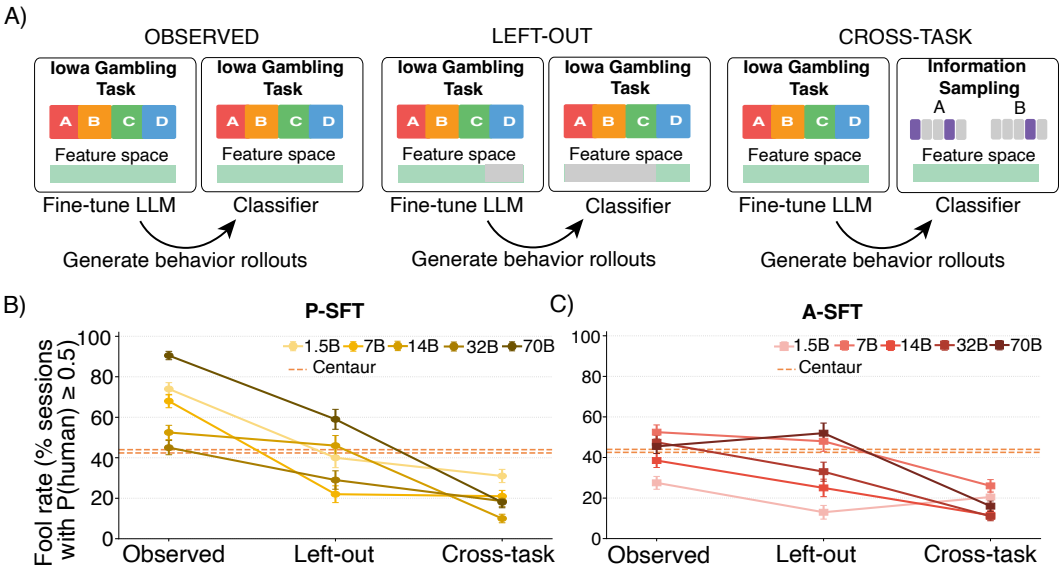


Figure 3: **Human-likeness from process-supervised fine-tuning is specific to the observed (supervised) features and task**. A) Three evaluation regimes for fine-tuned models. Observed: rollouts are generated on the training task and evaluated on the same process features used during fine-tuning. Left-out: rollouts on the training task are evaluated on left-out process features never used in fine-tuning. Cross-task: the fine-tuned model generates rollouts on a different task (e.g., fine-tuned on the Iowa Gambling Task, evaluated on Information Sampling), scored on that task's features. The percentage of sessions a Random Forest classifier scores as human (P(human) ≥ 0.5) for P-SFT (B) and A-SFT (C) at each model size, across the three regimes. Error bars show binomial SEM; the dashed line marks Centaur (44% on matched observed features, 43% on matched left-out features). P-SFT achieves high fool rates on observed features (up to 90.5% at 70B), but performance degrades on left-out features and collapses to 10–31% cross-task at every size, indicating that process alignment partially transfers to unmeasured features within the same task but does not generalize reliably to novel tasks.

representation was directly accessible/observed. However, this advantage diminished for left-out features and largely disappeared under cross-task evaluation, indicating that matching a known process representation does not necessarily produce more general human-like behavior.

Together, these results suggest that process provides more discriminative power than outputs across current AI models, and that broader human-process alignment will require representations that capture transferable structure across tasks rather than task-specific feature matching alone. Such representations could take the form of latent behavioral embeddings, hierarchical behavioral abstractions, or multi-task objectives that capture shared structure across different cognitive domains.

Limitations. Several limitations remain. First, our human data were drawn from a relatively small participant sample from a single pool. We therefore operationalize "human-like" process relative to the empirical behavioral distribution of this sampled population under the chosen feature representation, rather than as a claim about any canonical form of human cognition. Second, our fine-tuning evaluation is restricted to a limited subset of our original task battery, as we aimed to benchmark the methods against Centaur which only accepts text data in a formatted trial structure. Accordingly, it remains unclear whether similar process-alignment dynamics will hold for richer interactive environments. Lastly, process-based discrimination need not depend on a fixed battery of behavioral probes. Automated task generation and iterative task redesign could continuously introduce novel behavioral demands and target dimensions along which human and machine processes remain distinct, turning process-based evaluation into a dynamic red-teaming framework in which increasingly human-like models are evaluated against new process-level challenges rather than a static set of known behavioral signatures.

REFERENCES

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional AI: Harmlessness from AI Feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Peter W Battaglia, Jessica B Hamrick, and Joshua B Tenenbaum. Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45):18327–18332, 2013.

Antoine Bechara. Neurobiology of decision-making: risk and reward. In *Seminars in Clinical Neuropsychiatry*, volume 6, pp. 205–216, 2001.

Marcel Binz, Elif Akata, Matthias Bethge, Franziska Brändle, Fred Callaway, Julian Coda-Forno, Peter Dayan, Can Demircan, Maria K Eckstein, Noémi Éltető, et al. A foundation model to predict and capture human cognition. *Nature*, 644(8078):1002–1009, 2025.

Ned Block. Psychologism and Behaviorism. *The Philosophical Review*, 90(1):5–43, 1981.

Frederick Callaway, Falk Lieder, Paul M Krueger, and Thomas L Griffiths. Mouselab-MDP: A new paradigm for tracing how people plan. In *The 3rd Multidisciplinary Conference on Reinforcement Learning and Decision Making, Ann Arbor, MI*, 2017.

Murray Campbell, A Joseph Hoane Jr, and Feng-hsiung Hsu. Deep Blue. *Artificial Intelligence*, 134 (1-2):57–83, 2002.

Samuel J Cheyette and Steven T Piantadosi. A unified account of numerosity perception. *Nature Human Behaviour*, 4(12):1265–1272, 2020.

Luke Clark, Trevor W Robbins, Karen D Ersche, and Barbara J Sahakian. Reflection Impulsivity in Current and Former Substance Users. *Biological Psychiatry*, 60(5):515–522, 2006.

Anne GE Collins and Michael J Frank. How much of reinforcement learning is working memory, not reinforcement learning? A behavioral, computational, and neurogenetic analysis. *European Journal of Neuroscience*, 35(7):1024–1035, 2012.

Philip Michael Corsi. Human Memory and the Medial Temporal Region of the Brain. 1972.

Nelson Cowan. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, 24(1):87–114, 2001.

Stanislas Dehaene and Jean-Pierre Changeux. The Wisconsin Card Sorting Test: Theoretical Analysis and Modeling in a Neuronal Network. *Cerebral Cortex*, 1(1):62–79, 1991.

Miguel P Eckstein. Visual search: A retrospective. *Journal of Vision*, 11(5):14–14, 2011.

Paul M Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47(6):381, 1954.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Lu Wang, Weizhu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. *arXiv preprint arXiv:2106.09685*, 2021.

Mehrdad Jazayeri and Michael N Shadlen. Temporal context calibrates interval timing. *Nature Neuroscience*, 13(8):1020–1026, 2010.

George Jewell and Mark E McCourt. Pseudoneglect: a review and meta-analysis of performance factors in line bisection tasks. *Neuropsychologia*, 38(1):93–110, 2000.

Cameron R Jones and Benjamin K Bergen. Large Language Models Pass the Turing Test. *arXiv preprint arXiv:2503.23674*, 2025.

Stefan Künzell, Dominicus Sießmeir, and Harald Ewolds. Validation of the Continuous Tracking Paradigm for Studying Implicit Motor Learning. *Experimental Psychology*, 2017.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking Clever Hans predictors and assessing what machines really learn. *Nature Communications*, 10(1):1096, 2019.

Steven J Luck and Edward K Vogel. The capacity of visual working memory for features and conjunctions. *Nature*, 390(6657):279–281, 1997.

Nicholas P Maxwell, Mark J Huff, and Jacob M Namias. Predictive alternating runs and random task-switching sequences produce dissociative switch costs in the Consonant–Vowel/Odd–Even task. *Cognitive Processing*, 26(1):157–170, 2025.

James E Mazur. An Adjusting Procedure for Studying Delayed Reinforcement. In *The Effect of Delay and of Intervening Events on Reinforcement Value*, pp. 55–73. Psychology Press, 2013.

David Navon. Forest before trees: The precedence of global features in visual perception. *Cognitive Psychology*, 9(3):353–383, 1977.

Martin Nowak and Karl Sigmund. A strategy of win-stay, lose-shift that outperforms tit-for-tat in the Prisoner's Dilemma game. *Nature*, 364(6432):56–58, 1993.

Martin A Nowak, Karen M Page, and Karl Sigmund. Fairness Versus Reason in the Ultimatum Game. *Science*, 289(5485):1773–1775, 2000.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022.

Adrian M Owen, Kathryn M McMillan, Angela R Laird, and Ed Bullmore. N-back working memory paradigm: A meta-analysis of normative functional neuroimaging studies. *Human Brain Mapping*, 25(1):46–59, 2005.

Peter S Park, Simon Goldstein, Aidan O'Gara, Michael Chen, and Dan Hendrycks. AI deception: A survey of examples, risks, and potential solutions. *Patterns*, 5(5), 2024.

Andreas Plesner, Tobias Vontobel, and Roger Wattenhofer. Breaking reCAPTCHAv2. In *2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 1047–1056. IEEE, 2024.

Michael I Posner. Orienting of attention: Then and now. *The Quarterly Journal of Experimental Psychology*, 69(10):1864–1875, 2016.

Zenon W Pylyshyn and Ron W Storm. Tracking multiple independent targets: Evidence for a parallel tracking mechanism. *Spatial Vision*, 3(3):179–197, 1988.

Patrick M. A. Rabbitt. Errors and error correction in choice-response tasks. *Journal of Experimental Psychology*, 71(2):264–272, 1966.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

Jason Rajsic and Daryl E Wilson. Asymmetrical access to color and location in visual working memory. *Attention, Perception, & Psychophysics*, 76(7):1902–1913, 2014.

Kenneth Wingate Regan, Tamal Biswas, and Jason Zhou. Human and Computer Preferences at Chess. In *MPREF@AAAI*, 2014.

Bruno H Repp. Sensorimotor synchronization: A review of the tapping literature. *Psychonomic Bulletin & Review*, 12(6):969–992, 2005.

Kimron L Shapiro, Jane E Raymond, and Karen M Arnell. The attentional blink. *Trends in Cognitive Sciences*, 1(8):291–296, 1997.

Roger N Shepard and Jacqueline Metzler. Mental Rotation of Three-Dimensional Objects. *Science*, 171(3972):701–703, 1971.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359, 2017.

J Ridley Stroop. Studies of interference in serial verbal reactions. *Journal of experimental psychology*, 18(6):643, 1935.

Gábor J Székely and Maria L Rizzo. Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143(8):1249–1272, 2013.

Qwen Team. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*, 2024.

Nikolaus F Troje. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of Vision*, 2(5):2–2, 2002.

Alan M. Turing. Computing Machinery and Intelligence. *Mind*, 59(236):433–460, 1950.

Luis Von Ahn, Manuel Blum, Nicholas J Hopper, and John Langford. CAPTCHA: Using hard AI problems for security. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 294–311. Springer, 2003.

David Wechsler. *Manual for the Wechsler Adult Intelligence Scale*. The Psychological Corporation, New York, 1955.

Laura Weidinger, Jonathan Uesato, Maribeth Rauh, Conor Griffin, Po-Sen Huang, John Mellor, Amelia Glaese, Myra Cheng, Borja Balle, Atoosa Kasirzadeh, et al. Taxonomy of Risks posed by Language Models. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pp. 214–229, 2022.

Yifan Yao, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Zhibo Sun, and Yue Zhang. A survey on large language model (LLM) security and privacy: The Good, The Bad, and The Ugly. *High-Confidence Computing*, 4(2):100211, 2024.

A Appendix

A.1 Task Battery Catalog

Table 1: Full task battery process-feature catalog with task descriptions. CAPTCHA image task is split into two rows for Classic and Cross-Tile variants.

Task	Task description	Task source	Task-specific features
Attentional blink	Rapid serial visual presentation; detect two targets with varying temporal lag	Shapiro et al. (1997)	`blink_magnitude`, `t2_short_lag_acc`, `t2_long_lag_acc`, `t2_given_t1_acc`, `conditional_blink`
Biological motion	Point-light walker animation; judge walking direction	Troje (2002)	`direction_bias`, `acc_left`, `acc_right`, `acc_variability`
Classic CAPTCHA	Select all tiles matching a target object	Von Ahn et al. (2003)	`overselection_ratio`, `sequential_score`, `direction_change_rate`
Cross-Tile	Objects span tile boundaries, requiring spatial grouping	Von Ahn et al. (2003)	`overselection_ratio`, `sequential_score`, `direction_change_rate`

Task	Task Description	Task Source	Task-Specific Features
Change detection	Detect changes between alternating images	Luck & Vogel (1997)	`mean_flicker_cycles`, `flicker_cv`
Corsi	Spatial span task; reproduce sequence of block locations	Corsi (1972)	`acc_slope_span`, `transposition_rate`, `intrusion_rate`, `serial_first_acc`, `serial_middle_acc`, `serial_last_acc`, `primacy_recency`
Delay discounting	Choose between smaller immediate vs larger delayed rewards	Mazur (2013)	`discount_slope`, `choice_consistency`, `log_discount_k`
Digit span	Verbal working memory; recall digit sequences in order	Wechsler (1955)	`acc_slope_span`, `transposition_rate`, `intrusion_rate`, `serial_first_acc`, `serial_middle_acc`, `serial_last_acc`, `primacy_recency`
Fitts	Pointing task; click targets of varying size and distance	Fitts (1954)	`fitts_slope`, `fitts_intercept`, `click_precision_mean`, `click_precision_std`, `click_distance_px_mean`, `click_distance_px_std`
Iowa Gambling Task	Sequential card choices from decks with different reward-risk profiles	Bechara (2001)	`early_exploration`, `learning_slope`, `win_stay`, `lose_shift`, `stickiness`, `deck_entropy`
Line bisection	Mark midpoint of a line (spatial bias)	Jewell & McCourt (2000)	`mean_abs_error`, `error_slope_length`
Mental rotation	Judge whether rotated shapes are same or mirrored	Shepard & Metzler (1971)	`rotation_slope`, `angle_acc_delta`, `acc_angle_small`, `acc_angle_large`, `mirror_acc_effect`
MOT	Track moving targets among distractors	Pylyshyn & Storm (1988)	`load_slope`
Mouselab MDP	Explore states via mouse movements to maximize reward	Callaway et al. (2017)	`edge_fraction`, `mean_dist_to_edge`, `turn_rate`, `edge_hovers_mean`, `state_hovers_mean`, `unique_edges_explored`, `exploration_ratio`, `pre_move_hovers`
Navon	Global-local letters; identify large vs small letter under congruency manipulations	Navon (1977)	`navon_congruency_effect`, `global_advantage`, `navon_global_cong_acc`, `navon_global_incong_acc`, `navon_local_cong_acc`, `navon_local_incong_acc`, `navon_global_congruency_acc`, `navon_local_congruency_acc`, `global_acc_advantage`
N-back	Working memory task; detect repeats from N trials back	Owen et al. (2005)	`criterion`, `hit_rate`, `fa_rate`
Numerosity	Estimate which of two sets has more items	Cheyette & Piantadosi (2020)	`difficulty_effect`, `response_bias`, `distance_acc_slope`
Posner	Cueing task; respond to targets with valid/invalid spatial cues	Posner (2016)	`validity_acc_effect`, `valid_acc`, `invalid_acc`

Task	Task Description	Task Source	Task-Specific Features
Target pursuit	Track moving target with cursor; continuous control task	Künzell et al. (2017)	`cursor_smoothness`, `directional_accuracy`, `correction_rate`, `anticipation_lag_ms`
Rhythm tapping	Tap in time with rhythm; measures timing variability and drift	Repp (2005)	`tap_error_mean`, `tap_error_std`, `tap_drift`, `interval_cv`
Sampling	Sequential information sampling before making a decision	Clark et al. (2006)	`mean_total_samples`, `var_total_samples`, `mean_sample_bias`, `samples_easy`, `samples_medium`, `samples_hard`, `neutral_first_rate`, `effort_accuracy_corr`
Sequence prediction	Predict next element in a sequence with varying complexity	[N/A] source	`difficulty_slope`, `seq_acc_d1`, `seq_acc_d2`, `seq_acc_d3`, `seq_acc_d4`, `seq_acc_d5`
Slingshot	Aim and launch projectile to hit targets; physics-based control task	Battaglia et al. (2013)	`mean_aim_time`, `aim_time_cv`, `distance_calibration_r`, `mean_angle_delta_45`, `angle_variability`, `undershoot_rate`, `overshoot_rate`
Stroop	Name ink color of words; congruent vs incongruent trials	Stroop (1935)	`post_error_slowing`, `congruency_acc_effect`, `cong_acc`, `incong_acc`
Task switching	Alternate between task rules; measures switching cost	Maxwell et al. (2025)	`switch_cost_acc`, `repeat_acc`, `switch_acc`
Temporal reproduction	Reproduce time intervals; measures temporal bias and variability	Jazayeri & Shadlen (2010)	`temporal_bias`, `mean_ratio`, `regression_to_mean`
Tower physics	Predict stability of stacked objects (intuitive physics)	Battaglia et al. (2013)	`complexity_slope`
Ultimatum game	Fairness decision-making	Nowak et al. (2000)	`responder_accept_rate`, `reject_unfair_rate`
Visual search	Find target among distractors	Eckstein (2011)	`search_slope`, `search_click_precision`, `click_dist_setsize_slope`
Visual Working Memory	Recall location/color under varying set sizes	Rajsic & Wilson (2014)	`error_slope_setsize`, `bias_x_mean`, `bias_y_mean`, `bias_radius_mean`, `bias_anisotropy`, `large_error_frac`, `swap_error_rate`
WCST	Infer and update sorting rule from feedback	Dehaene & Changeux (1991)	`learning_slope`, `post_error_slowing`, `shift_error_rate`, `perseveration_cost`, `rule_acc_variability`

A.2 Full task results

Table 2: Performance across combinations of observed and left-out features. The results show that the reduction in the human-agent process gap is attenuated when evaluation features are not observed during training.

Combination	Feature set	P-SFT	A-SFT
1	Observed features	0.06	0.62
	Left-out features	0.32	0.66
2	Observed features	0.13	0.40
	Left-out features	0.35	0.44
3	Observed features	0.08	0.47
	Left-out features	0.321	0.55

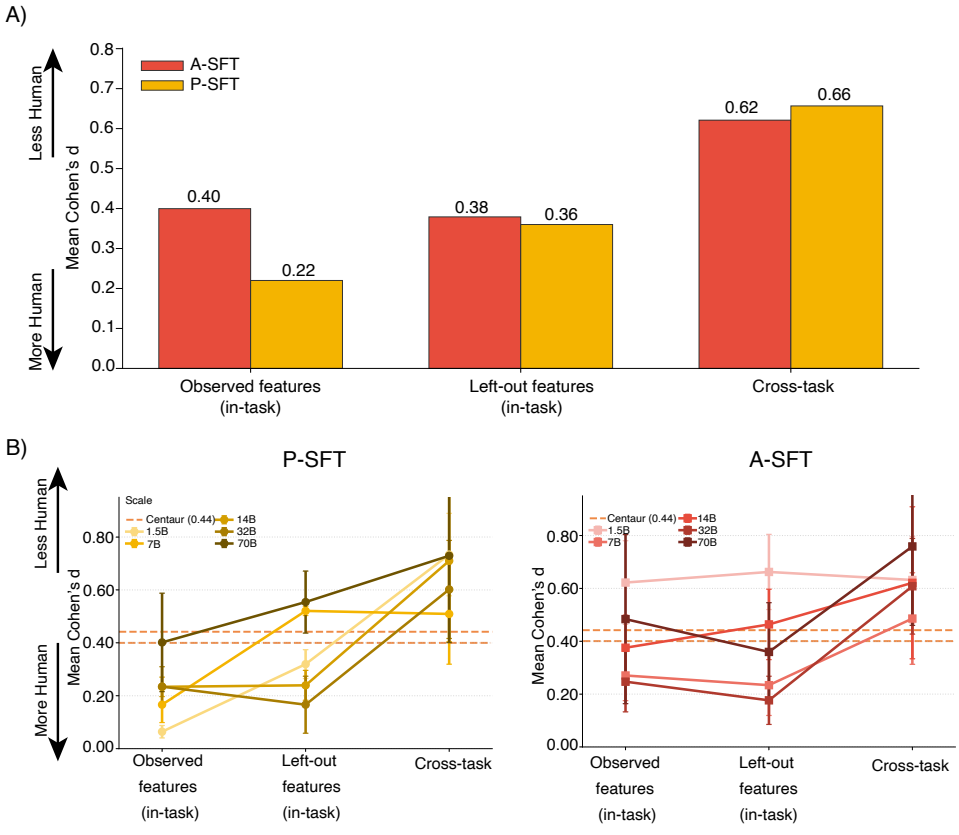


Figure 4: P-SFT models show lower overall distance to human process distributions than A-SFT models. This advantage diminishes when evaluated on left-out features from the same task and largely disappears under cross-task evaluation.

Table 3: Distributional distance to humans measured by mean absolute Cohen's d across process features (lower is more human-like). Centaur more closely matched human process distributions than frontier agents across all evaluated tasks.

Method	Sampling	IGT	WCST	Average
Base Qwen (1.5B)	0.89	0.81	0.88	0.86
Claude Sonnet	1.15	3.32	0.48	1.65
GPT-5	0.27	1.91	1.91	1.36
Gemini 2.5 Pro	1.26	0.56	1.38	1.07
Centaur (70B)	0.15	0.42	0.76	0.44
A-SFT (1.5B)	0.51	0.42	0.93	0.62
A-SFT (7B)	0.17	0.18	0.46	0.27
A-SFT (14B)	0.27	0.23	0.62	0.38
A-SFT (32B)	0.47	0.11	0.15	0.25
A-SFT (70B)	0.10	0.24	1.12	0.48
P-SFT (1.5B)	**0.11**	**0.06**	**0.03**	**0.06**
P-SFT (7B)	0.07	0.14	0.30	0.17
P-SFT (14B)	0.30	0.22	0.18	0.23
P-SFT (32B)	0.37	0.11	0.23	0.23
P-SFT (70B)	0.76	0.13	0.31	0.40
Left-out features				
Centaur (70B)	0.25	0.48	0.48	0.40
A-SFT (1.5B)	0.68	0.41	0.90	0.66
A-SFT (7B)	0.17	0.08	0.46	0.23
A-SFT (14B)	0.34	0.32	0.73	0.46
A-SFT (32B)	0.36	0.07	0.10	0.18
A-SFT (70B)	0.17	0.17	0.73	0.36
P-SFT (1.5B)	0.39	0.36	0.21	0.32
P-SFT (7B)	0.51	0.55	0.50	0.52
P-SFT (14B)	0.35	0.21	0.16	0.24
P-SFT (32B)	0.38	0.09	0.03	0.17
P-SFT (70B)	0.78	0.51	0.38	0.55
Cross-task				
A-SFT (1.5B)	0.74	0.31	0.84	0.63
A-SFT (7B)	0.29	0.33	0.83	0.49
A-SFT (14B)	0.69	0.09	1.08	0.62
A-SFT (32B)	0.61	0.29	0.92	0.61
A-SFT (70B)	0.59	0.34	1.34	0.76
P-SFT (1.5B)	1.00	0.46	0.73	0.73
P-SFT (7B)	0.28	0.36	0.89	0.51
P-SFT (14B)	0.76	0.16	1.21	0.71
P-SFT (32B)	0.61	0.28	0.92	0.60
P-SFT (70B)	0.54	0.28	1.37	0.73

Table 4: Combined classifier fool rate. A Random Forest classifier (AUC = 0.89 $\pm$ 0.03) was trained to distinguish humans from off-the-shelf LLM agents using process features concatenated across all three tasks. Test set off-the-shelf runs, fine-tuned models and Centaur were held out from training and evaluated against the fitted classifier. Higher fool rate indicates more human-like behavior.

Method	Fool rate ↑	Mean P(human)
Off-the-shelf models (held-out evaluation runs)		
GPT-5	7.6%	0.18
Gemini 2.5 Pro	0.0%	0.16
Claude Sonnet	0.0%	0.05
Base Qwen (1.5B)	6.8%	0.26
Base Qwen (7B)	17.4%	0.34
Base Qwen (14B)	8.6%	0.27
Base Qwen (32B)	3.4%	0.26
Base Llama (70B)	18.4%	0.33
Centaur (70B)	44.0%	0.45
A-SFT (1.5B)	27.5%	0.37
A-SFT (7B)	52.5%	0.50
A-SFT (14B)	38.5%	0.43
A-SFT (32B)	47.5%	0.50
A-SFT (70B)	45.5%	0.48
P-SFT (1.5B)	74.0%	0.64
P-SFT (7B)	68.0%	0.62
P-SFT (14B)	52.5%	0.52
P-SFT (32B)	45.0%	0.47
P-SFT (70B)	**90.5%**	**0.73**
Left-out features		
Centaur (70B)	43.0%	0.48
A-SFT (1.5B)	13.0%	0.30
A-SFT (7B)	48.0%	0.50
A-SFT (14B)	25.0%	0.36
A-SFT (32B)	33.0%	0.44
A-SFT (70B)	52.0%	0.54
P-SFT (1.5B)	40.0%	0.46
P-SFT (7B)	22.0%	0.36
P-SFT (14B)	46.0%	0.45
P-SFT (32B)	29.0%	0.40
P-SFT (70B)	59.0%	0.54
Cross-task		
A-SFT (1.5B)	20.5%	0.33
A-SFT (7B)	26.0%	0.36
A-SFT (14B)	11.5%	0.27
A-SFT (32B)	11.0%	0.30
A-SFT (70B)	16.0%	0.31
P-SFT (1.5B)	31.0%	0.39
P-SFT (7B)	21.0%	0.33
P-SFT (14B)	10.0%	0.26
P-SFT (32B)	18.6%	0.33
P-SFT (70B)	18.0%	0.34

Table 5: Energy distance—a multivariate distributional metric defined as $2\,\mathbb{E}\|X - Y\| - \mathbb{E}\|X - X'\| - \mathbb{E}\|Y - Y'\|$, computed on per-feature human-standardized features—captures both location and shape differences between a method's and humans' joint feature distributions; a value of zero indicates identical distributions.

Method	**Sampling**	**IGT**	**WCST**	**Average**
Base Qwen (1.5B)	1.50	1.27	1.04	1.27
Claude Sonnet	1.33	11.14	0.61	4.36
GPT-5	0.29	5.84	2.28	2.81
Gemini 2.5 Pro	2.06	1.30	1.42	1.59
Observed features				
Centaur (70B)	0.03	0.52	0.56	0.37
A-SFT (1.5B)	0.55	0.53	0.97	0.68
A-SFT (7B)	0.07	0.16	0.21	0.15
A-SFT (14B)	0.16	0.16	0.47	0.26
A-SFT (32B)	0.42	0.10	0.04	0.19
A-SFT (Llama 70B)	0.07	0.14	1.04	0.42
P-SFT (1.5B)	**0.11**	**0.06**	**0.06**	**0.07**
P-SFT (7B)	0.09	0.16	0.13	0.13
P-SFT (14B)	0.16	0.14	0.10	0.14
P-SFT (32B)	0.23	0.05	0.07	0.12
P-SFT (Llama 70B)	0.48	0.15	0.10	0.24
Left-out features				
Centaur (70B)	0.15	0.29	0.16	0.20
A-SFT (1.5B)	1.24	0.26	0.60	0.70
A-SFT (7B)	0.12	0.03	0.20	0.11
A-SFT (14B)	0.45	0.13	0.44	0.34
A-SFT (32B)	0.45	0.03	0.02	0.17
A-SFT (70B)	0.14	0.02	0.40	0.18
P-SFT (1.5B)	0.41	0.33	0.04	0.26
P-SFT (7B)	1.50	0.71	0.24	0.81
P-SFT (14B)	0.32	0.07	0.03	0.14
P-SFT (32B)	0.45	0.04	0.01	0.17
P-SFT (70B)	1.15	0.57	0.09	0.60
Cross-task				
A-SFT (1.5B)	1.05	0.39	0.83	0.76
A-SFT (7B)	0.23	0.38	0.73	0.45
A-SFT (14B)	0.78	0.08	1.13	0.66
A-SFT (32B)	0.61	0.24	0.89	0.58
A-SFT (70B)	0.54	0.29	1.54	0.79
P-SFT (1.5B)	1.28	0.44	0.59	0.77
P-SFT (7B)	0.24	0.42	0.95	0.54
P-SFT (14B)	0.78	0.21	1.42	0.80
P-SFT (32B)	0.57	0.16	0.86	0.53
P-SFT (70B)	0.40	0.14	1.53	0.69

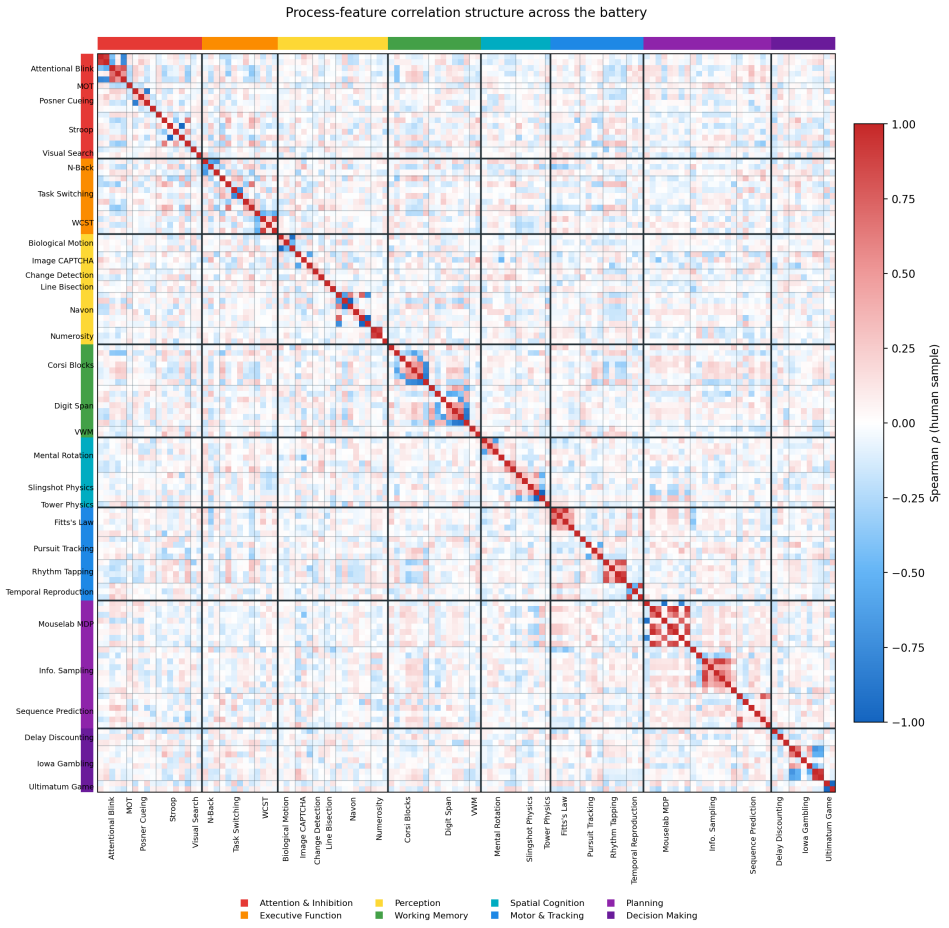


Figure 5: Correlation of features across the full task battery. Features across tasks show generally low correlations, indicating limited overlap in the measured process dimensions.

A.3 P-SFT: IMPLEMENTATION DETAILS

All fine-tuning was performed on a single NVIDIA H200 SXM GPU (141 GB VRAM). We fine-tuned Qwen2.5-Instruct (Team, 2024) models (1.5B, 7B, 14B, 32B) and Llama3.3-Instruct (70B) via LoRA (Hu et al., 2021) (rank 8, alpha 16, dropout 0.05) applied to all attention and MLP projection layers. Training used AdamW with learning rate 10^{-5}, gradient clipping at norm 1, and gradient checkpointing to fit the per-rollout state expansion in memory. A-SFT trained for 300 steps of cross-entropy on human actions. P-SFT added 30 steps of cross-entropy warmup followed by up to 150 steps of joint optimization, with early stopping when the feature-matching loss plateaued (patience of 20–50 steps depending on task). At each joint step, we sampled a fresh stimulus sequence from a random training participant's run, preventing the model from overfitting to a single sequence of task stimuli.

Evaluation. We use 2-fold cross-validation: 97 human participants are split into two folds (47 and 50). Each fold's model is trained on one half and evaluated against held-out humans from the other half. Per-fold rollouts (100 per fold, 200 total) are compared to per-fold held-out humans, and results are pooled. This ensures no overlap between training and evaluation participants.

Information Sampling. This task involves multiple decisions per trial: the agent chooses to reveal a tile from option A, reveal a tile from option B, or stop and choose an option. The run-level features are mean and variance of the per-trial total tile count. Because each trial involves a variable-length sequence of decisions, we compute the expected tile count differentiably by expanding the set of reachable game states at each within-trial step, weighted by the policy's action probabilities. States with identical observable histories (e.g., two A-tiles revealed via different orderings) are merged. The expected stopping time is then

$$\mathbb{E}[T] = \sum_{t=0}^{T_{\max}} t \cdot P(\text{stop at step } t), \tag{2}$$

computed exactly by accumulating $P(\text{stop} \mid \text{state}_t) \cdot P(\text{state}_t)$ over the expansion. We cap the frontier at 16 active states per step, which exceeds the theoretical maximum of 11 distinct observable states after merging (since with 5 tiles per option, the merged state space at step t is the set of (n_A,n_B) pairs with $n_A + n_B = t$). At trial boundaries, we select the most-likely outcome from the frontier (greedy stop time and choice) to construct the prompt continuation for subsequent trials, matching the prompt format used at evaluation time.

Iowa Gambling Task. Each trial is a single 4-way choice over decks A, B, C, D; no within-trial state expansion is needed. The six run features, all differentiable in the model's action probabilities p_t, are:

- *learning_slope*: mean good-deck probability in the second half minus the first half, $\frac{1}{\lfloor T/2 \rfloor} \sum_{t \geq \lceil T/2 \rceil} p_t^{\text{good}} - \frac{1}{\lceil T/2 \rceil} \sum_{t < \lceil T/2 \rceil} p_t^{\text{good}}$, where $p_t^{\text{good}} = p_t(C) + p_t(D)$
- *stickiness*: $\frac{1}{T-1} \sum_{t=2}^{T} p_t(a_{t-1})$, the mean probability assigned to the previous trial's sampled action (the sampled action a_{t-1} is treated as fixed; gradients flow through p_t only)
- *deck_entropy*: $H(\bar{p}) = -\sum_d \bar{p}_d \log \bar{p}_d$, where $\bar{p} = \frac{1}{T} \sum_t p_t$ is the mean action distribution across trials
- *win_stay*: mean probability of repeating the previous action on trials following a positive outcome, $\frac{1}{|\mathcal{W}|} \sum_{t \in \mathcal{W}} p_t(a_{t-1})$, where $\mathcal{W} = \{t : \text{net}_{t-1} > 0\}$
- *lose_shift*: mean probability of switching away from the previous action on trials following a non-positive outcome, $\frac{1}{|\mathcal{L}|} \sum_{t \in \mathcal{L}} (1 - p_t(a_{t-1}))$, where $\mathcal{L} = \{t : \text{net}_{t-1} \leq 0\}$
- *good_deck_rate*: $\frac{1}{T} \sum_t (p_t(C) + p_t(D))$

Win/loss classification at each trial is determined by the sampled action and the deterministic deck outcomes, and is treated as fixed for gradient computation. Gradients flow only through the probability vectors p_t.

Wisconsin Card Sorting Task. Each trial is a single 4-way choice (match to reference card 1, 2, 3, or 4). Whether the choice is correct depends on the current hidden rule and the test card. The two run features are:

- *perseveration_cost*: mean $p(\text{correct})$ on non-shift trials minus mean $p(\text{correct})$ on shift trials, where $p(\text{correct}) = p_t(\text{correct index})$ is the probability the model assigns to the correct card on trial t
- *learning_slope*: mean $p(\text{correct})$ in the second half of the run minus the first half

Both are direct functions of the model's per-trial softmax probabilities evaluated at the correct action index, which is known from the stimulus sequence.

Training stability. Under the closed-form feature estimator, the feature-matching loss decreases monotonically during phase-2 training across all three tasks, typically reaching a plateau around step 90-100. We use AdamW with learning rate 10^{-5}, gradient clipping at norm 1, and gradient checkpointing to fit the per-rollout computation in memory.

A.4 Task Instructions and Agent Prompts

A.4.1 Information Sampling

Agent Prompt (Sampling Phase)

There are two rows of tiles:
- Option A (top/blue): 5 tiles
- Option B (bottom/red): 5 tiles

Hidden tiles show "?". Revealed tiles show numbers (0–100). Your goal is to determine which option has the higher average.
Decide whether to sample another tile or stop and choose.
If sampling, respond with: `{"action": "sample", "option": "A", "tile": <0-4>}`
or
`{"action": "sample", "option": "B", "tile": <0-4>}`
If ready to choose: `{"action": "choose"}`

Agent Prompt (Choice Phase)

Based on the revealed tile values, which option has the higher average?
Respond with ONLY:
`{"choice": "A"}` or `{"choice": "B"}`

Participant Instructions

You will see two rows of 5 hidden tiles: Option A (top/blue) and Option B (bottom/red).
Each tile hides a number. One option tends to have higher values than the other.
Click a tile to reveal its value. Each flip costs points.
Choosing correctly earns bonus points.
When you are confident, click "I'm ready to choose."
Your goal is to determine which option has higher values overall while minimizing flip costs.

A.4.2 Visual Working Memory

Agent Prompt

Image 1 shows colored squares during encoding. Memorize each color's position.
Image 2 shows an empty canvas and a target color labeled "Click where this color was."
Find where the target color was located and report its center coordinates.
Respond with ONLY:
`{"x": <number>, "y": <number>}`

Participant Instructions

You will see several colored squares flash on the screen.
After they disappear, one of the colors will appear as the target.
Click where on the screen that color was previously shown.

A.4.3 VISUAL SEARCH

Agent Prompt

Find the RED CIRCLE and click on it.
It will be hidden among red squares and blue circles.
Respond with ONLY:
`{"x": <number>, "y": <number>}`

Participant Instructions

Find the RED CIRCLE and click on it.
It will be hidden among red squares and blue circles.

A.4.4 MULTIPLE OBJECT TRACKING

Agent Prompt (Highlight Phase)

Some dots are highlighted as targets.
Report the coordinates of each highlighted target.
Respond with ONLY:
`{"targets": [{"x": N, "y": N}, ...]}`

Agent Prompt (Tracking Phase)

Track the previously highlighted dots as they move.
Update each target's position to the nearest dot.
Respond with ONLY:
`{"targets": [{"x": N, "y": N}, ...]}`

Participant Instructions

Some dots will flash yellow—these are your targets.
All dots will then begin moving. Track the targets.
When the dots stop, click the dots you believe were the targets.

A.4.5 DELAY DISCOUNTING

Agent Prompt

Look at the two monetary options on screen: A smaller amount now or A larger amount later
Determine the amounts and delay, reason about the tradeoff, and report your choice.
Respond as JSON:
`{"now_amount": N, "later_amount": N, "delay": "X", "choice": "now" or "later"}`

Participant Instructions

On each trial you will choose between:

- A smaller amount now
- A larger amount later

There are no right or wrong answers—choose whichever you prefer.