

Choosing LLMs for Strategic Agent Interactions

Arastun Mammadli
University of Exeter
United Kingdom
am2169@exeter.ac.uk

Herbert Woisetschl ger
IBM Research
Switzerland
h.woisetschlaeger@ibm.com

Jawad Fayaz
University of Exeter
United Kingdom
j.fayaz@exeter.ac.uk

Shiqiang Wang
University of Exeter
United Kingdom
s.wang9@exeter.ac.uk

Abstract

Game-theoretic benchmarks claim that large language model (LLM) strategic decision-making varies across game types, interaction structures, and capability dimensions. However, they only evaluate strategic performance across individual fixed models. In parallel, contemporary LLM routers pick models under quality-cost trade-offs, but with no notion of a strategic structure. They do not account for an opponent that might be unknown, and that can adapt over time and shift the feedback signal. To address this, we extend the game-theoretic evaluations and define strategic selection as the meta game layered on top of the base game. In each round, a strategic selector fields an LLM from a portfolio of models to navigate a game payoff matrix that is latent and learned online. We find that the evaluation signal is game-specific, and that extracting a reliable signal requires a much higher number of seed runs N compared to traditional benchmarking setups. Evaluating across 5 games and 10 LLMs, we show a model’s strategic competence varies significantly depending on the game setup and the opponent faced. Building on this, we implement a suite of selector methods and find that approaches that rely on query embeddings or LLM self-reflection do not consistently capture the underlying strategic signal. A lightweight multi-armed bandit algorithm performs more reliably across games, though it still falls short of the best fixed model. Finally, we show that simple non-stationary and adversarial bandit variants fail to outperform a stationary bandit baseline, pointing to a gap in how current methods handle a shifting opponent in strategic settings.

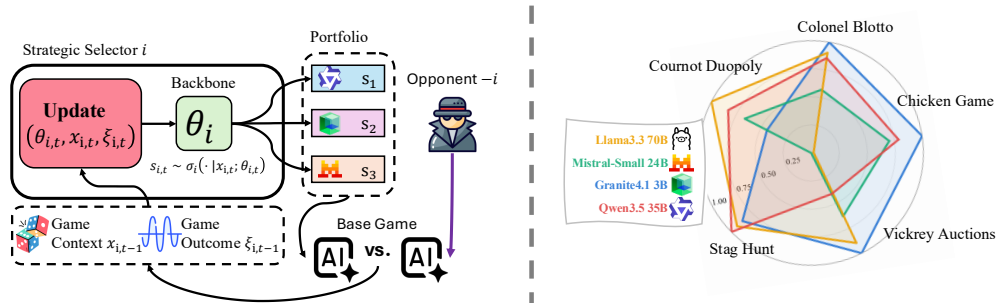


Figure 1: *Left:* A strategic selector i plays a repeated game over T rounds against an unknown opponent $-i$ and observes the game outcome $\xi_{i,t}$. The $\text{UPDATE}(\theta_{i,t}, x_{i,t}, \xi_{i,t})$ function is applied at every round t . *Right:* Strategic performance (game utility v) varies across games and models.

1 Introduction

Large language models (LLMs) are increasingly deployed in multi-agent settings as part of a larger compound system. However, a growing share of deployers treat the inter-agent dynamics, competitive and cooperative alike, as a black box. This makes it hard to build systems that remain capable, efficient, and reliable. Existing work addresses pieces of this problem in isolation. The strategic decision-making literature focuses on a fixed model’s play under a specified game setting [7, 26, 35]. In parallel, LLM routing addresses model selection under performance-cost trade-off [5, 24, 28], but with a static task distribution and no notion of a strategic structure or opponent. Neither accounts for the fact that contemporary LLM portfolios differ not only in scale and cost, but also in the type and quality of strategic behaviour they exhibit.

Our work reframes model selection itself as a strategic decision. Concretely, we extend the game-theoretic evaluations and define strategic selection as the opponent-conditioned meta game layered on top of the base game. Here the base game is played between the fielded LLMs, while the meta game is played between two strategic selectors that field a model of their choice each round. In repeated settings, the dynamic and adaptive nature of this problem (as opposed to traditional routers operating on a fixed distribution of tasks) means our selection should be strategic. Beyond balancing the performance-cost trade-offs, such a selection layer can help steer the deployer’s general play, adding control over the reliability and safety of the interaction (Figure 1). We ask the question:

Given LLM strategic capabilities vary across different games and models, can a selector exploit this by learning which model to field against an unknown and possibly adaptive opponent?

Why This Matters. Traditional routers operate under the assumption of a fixed task distribution, learning a static mapping from queries to optimal models. In contrast, operating against an adaptive opponent transforms this into a non-stationary learning problem, where the optimal strategy can continuously shift over time. A further challenge arises from the partial observability of the meta game’s payoff matrix. In practice, a selector cannot observe the opponent’s net gain, since the opponent’s model choice is unknown, nor can it observe the counterfactual payoffs of fielding an alternative model in the current game state. We also acknowledge that LLMs are stochastic models. In strategic selection, we introduce an additional layer of environmental noise through the dynamic responses of the opponent. Therefore, benchmarking and result reporting call for a careful statistical treatment. Our contributions are threefold:

- **Strategic Model Selection.** We formalise strategic selection as a repeated general-sum meta game between two players. Beyond capability and cost, this lets us treat the interaction itself as a first-class design consideration for model selection.
- **Quantifying Noise in Strategic Play.** We characterise the signal and noise levels across games, and suggest adapting methods from traditional hypothesis testing to decide on the number of seeds per a game-portfolio pair.
- **Empirical Investigation.** To set up our empirical investigation, we select 10 models from different families, and adapt a range of existing routing methods (Table 1). We first show that LLM strategic capabilities are scenario-dependent and exploitable by a selector. We then investigate the performance of adapted methods against both fixed models and selector opponents.

2 Related Work

2.1 Strategic LLM Decision-Making

A growing literature explores LLMs in strategic settings (e.g. how LLMs play games against each other). Game-theoretic benchmarks [7, 10, 26, 33] and frameworks [11, 31, 34, 35] test LLMs in normal-form games and find that performance is uneven and structure-dependent. Others [3, 23, 30, 32] evaluate capability on a higher level (e.g. planning, collaboration, theory of mind) and consistently find spiky, uneven profiles across models. Similarly, Mei et al. [13] and Horton et al. [9] run behavioural Turing tests and show that different models exhibit different strategic dispositions (i.e. behavioural profiles). Motivated by such findings, Raman et al. [19] outlines the need for an element-level (model, prompt, domain) selection approach, but leaves it to future work. We note that despite extensive coverage, the existing literature always fixes the model per player.

Table 1: Implemented suite of selector methods. *Cost-Aware*: the method accounts for a quality-cost trade-off when selecting a model. *Opp.-ID*: the opponent’s identity is revealed to the selector. *Online*: the method learns online (round-by-round) from game outcomes. *Embed.*: the method encodes context into embeddings before feeding them into the predictor. *Non-Stat.*: the method is designed to handle a non-stationary environment.

Method	Cost-Aware	Opp.-ID	Online	Embed.	Non-Stat.	Source
Fixed Model	×	×	×	×	×	—
Per-Opponent Full-Information	✓	✓	×	×	×	—
<i>Implemented selectors</i>						
LLM Selector	✓	×	✓	×	×	Shinn et al. [20]
RouterDC	✓	×	×	✓	×	Chen et al. [5]
Causal LLM Router	✓	×	✓	✓	×	Tsiourvas et al. [24]
Thompson Sampler	✓	×	✓	×	×	William et al. [27]
Discounted Thompson Sampler	✓	×	✓	×	✓	Raj and Kalyani [18]
Sliding-Window Thompson Sampler	✓	×	✓	×	✓	Raj and Kalyani [18]
EXP3	✓	×	✓	×	✓	Auer et al. [2]

2.2 LLM Routing

An orthogonal direction learns to predict a best-fit model for a given query, often trading off performance for cost and relying on minimal two-model setups [1, 6, 16]. Others extend this to incorporate more models [4, 17, 21, 29] or train on limited observational data [24, 28]. These works represent queries as vector embeddings to semantically capture their scope and difficulty. However, we argue that discriminative signal in strategic selection largely comes from evolving game history (actions and outcomes), and two consecutive rounds with near identical embeddings can call for opposite model choices. In the absence of routing methods that explicitly model this structure, we also consider bandit-style per-round feedback [2, 27] as a paradigm suited to a non-stationary interactive signal.

3 Strategic LLM Selection

3.1 What is Strategic Selection?

In multi-agent systems, a setting is *non-strategic* if the objective concerns only the overall performance of the system or its performance-cost trade-off. A setting is *strategic* if it also accounts for the interaction between agents, including how they communicate, coordinate, and negotiate. Modelling this interaction can produce systems that are more reliable and efficient. In a cooperative setting, it improves the coordination between agents [12]. In a competitive setting, it gives the deployer more control over the interaction with an opponent and helps steer the interaction toward the deployer’s preferences. For example, for a given business, the reward of the negotiation, the cost of using AI, and the consistency (i.e. associated risk) of the results can all matter at varying levels.

3.2 Problem Formulation

We formalise strategic LLM selection as a repeated, contextual, general-sum game with two layers: a *base game* that fielded models play, and a selection *meta game* that the deployers play. Strategic selection focuses on the meta layer, a two-player game with i indexing the player (deployer) and $-i$ its opponent. Repeated games are played over rounds $t = 1, \dots, T$, where each round t is associated with context $x_{i,t} \in \mathcal{X}$ (e.g. game setup, instructions, past history). Deployer i ’s set of pure strategies (meta game actions) S_i is defined as the set of models $\{1, \dots, M_i\}$ in its portfolio. In each round, i selects a model $s_{i,t} \sim \sigma_i(\cdot \mid x_{i,t})$, where $\sigma_i \in \Delta(S_i)$ is a mixed strategy (i.e. selection policy). Per-round context $x_{i,t}$ is then passed down to the fielded model to produce the base game action $s_{i,t}(x_{i,t}) \rightarrow a_{i,t}$. The base game (i.e. of fielded models $s_{i,t}$ against $s_{-i,t}$) produces a utility $v_{i,t}$ at an inference cost $c_{i,t}$ (tokens spent), as well as a net gain of $g_{i,t}$ that captures their trade-off. Together, player i ’s game outcome is denoted as $\xi_{i,t} = (s_{i,t}, a_{i,t}, g_{i,t})$ where the opponent’s model choice $s_{-i,t}$ remains hidden. Drawing from the online learning literature, each selector maintains an adaptive parameter vector $\theta_{i,t}$ that updates after each round based on observed context $x_{i,t}$ and game outcome $\xi_{i,t}$:

$$s_{i,t} \sim \sigma_i(\cdot \mid x_{i,t}; \theta_{i,t}), \quad \theta_{i,t+1} = \text{UPDATE}(\theta_{i,t}, x_{i,t}, \xi_{i,t}). \quad (1)$$

Any architecture that maps $\theta_{i,t}$ to a selection policy $\sigma_{i,t}$ and supports updates from online feedback qualifies as a backbone. Such options can include classical multi-armed bandit samplers [27], neural network-based classifiers [5, 24], or at the frontier, an LLM prompted with game history. Consequently, $\theta_{i,t}$'s definition depends on the backbone, such as the network weights for a classifier, or the per-arm posterior for a Thompson sampler. We note that the game outcome is also incorporated into the next round's context $x_{i,t+1} = f(x_{i,t}, \xi_{i,t})$ to be used in base game decision-making. Finally, this is repeated over N independent seed runs (where the parameter vector is reset for each seed $\theta_{i,1} \leftarrow \theta_i^0$) to account for stochastic variance in strategic play. Algorithm 1 summarises the overall procedure.

3.3 Noise in Strategic Play

Single-trial LLM decision-making is noisy, subject to the probabilistic sampling mechanism, and sensitive to variations in prompts. Related work addresses LLM evaluations by adapting cross-disciplinary methods such as signal processing [8], experiment-design theory [14], and psychometrics [15]. They find that in traditional evaluations (e.g. question-answering benchmarks), even a modest amount of replication (e.g. $N \approx 5$) substantially stabilises the estimates. We argue that due to its interactive opponent-conditioned structure, strategic LLM evaluation is inherently noisier and requires a higher number of replications. To quantify the noise in strategic play, we define a method to measure the levels of signal and noise across games by adapting Heineman et al. [8]'s work. Building on this, we develop approaches to choose N for a given game-portfolio pair [14]. See Appendix B for further details.

3.4 Evaluation Metrics

To measure strategic performance, we capture the game utility v_i , associated cost c_i , and their trade-off represented as the net gain g_i . The game utility v_i is min-max normalised ($\hat{v}_i \in [0, 1]$) based on its game-specific bounds (e.g. the economic profit in Cournot Duopoly, or the number of winning battlefields in Colonel Blotto game). The associated inference cost c_i is represented by the per-model token usage cost (in dollars \$) using the OpenRouter¹ listings as market proxies. This cost is min-max normalised ($\hat{c}_i \in [0, 1]$) based on the portfolio-bounds (minimum and maximum per-token price observed across the model portfolio). Finally, we define net gain g_i as the weighted sum of normalised utility $\hat{v}_i$ and cost $\hat{c}_i$

$$g_i = \frac{\hat{v}_i + \lambda(1 - \hat{c}_i)}{1 + \lambda} \quad g_i \in [0, 1], \quad (2)$$

where λ represents the cost trade-off weight and is fixed at $\lambda = 1$ across our runs. Any game episode is replicated across N seeds to account for stochastic noise. Each of the N seed runs yields a single sample, from which we compute the mean $\bar{X}$, the standard error (SE), and the 95% confidence interval (CI). Alternative metrics include the in-game winning rate (i.e. we ignore the raw g_i and measure $\mathbb{1}[g_i > g_{-i}]$), or representing the cost c_i by the per-model throughput (tokens produced per second).

4 Experiments

We provide empirical evidence in 3 steps and motivate the problem of strategic LLM selection. First, an exhaustive pairwise matchup across 10 models and 5 games motivates the problem, showing

¹<https://openrouter.ai/models>

Algorithm 1: Repeated strategic selection, de-ployer i 's view, in game $\mathcal{G}$, across N seed runs

```
//  $[\cdot]$  denotes quantities hidden from  $i$ 
1 Initialize  $S_i = \{1, \dots, M_i\}$ ,  $\theta_i^0$ ,  $T$  rounds,  $N$  seeds;
   for  $n \leftarrow 1$  to  $N$  do
     // runs are independent
2      $\theta_{i,1} \leftarrow \theta_i^0$ ,  $x_{i,1} \in \mathcal{X}$ ;
3     for  $t \leftarrow 1$  to  $T$  do
4          $s_{i,t} \sim \sigma_i(\cdot | x_{i,t}; \theta_{i,t})$ ;
           // opponent model not observed
5          $[s_{-i,t} \sim \sigma_{-i}(\cdot | x_{-i,t}; \theta_{-i,t})]$ ;
           // base game resolves
6          $a_{i,t} \leftarrow s_{i,t}(x_{i,t})$ ,  $a_{-i,t} \leftarrow [s_{-i,t}(x_{-i,t})]$ ;
7          $v_{i,t}, c_{i,t}, g_{i,t} \leftarrow \mathcal{G}(a_{i,t}, a_{-i,t})$ ;
8          $\xi_{i,t} \leftarrow (s_{i,t}, a_{i,t}, g_{i,t})$ ;
           // selector parameter is updated
9          $\theta_{i,t+1} \leftarrow \text{UPDATE}(\theta_{i,t}, x_{i,t}, \xi_{i,t})$ ;
10         $x_{i,t+1} \leftarrow f(x_{i,t}, \xi_{i,t})$ ;
```

that strategic capability is varied and exploitable by a selector. Following these findings, a suite of selectors is run against fixed but unknown models (i.e. opponent is a fixed LLM) to evaluate their potential as strategic selectors. Finally, these selectors are deployed against adaptive opponents to simulate the non-stationary dynamics of real-world settings. We note that reported results should not be compared across (Tables 2 to 4), as a method’s performance is a function of the specific opponent pool it faced which varies across settings (e.g. the runs against adaptive opponents do not include fixed models).

4.1 Design

Models and Games. The model portfolio draws on 10 open-weight LLMs across four families (Granite, Mistral, Llama, and Qwen) ranging from 3B to 70B (Appendix C.1) to capture varying capability levels, usage costs, and dispositions. The game inventory captures 5 classical game-theoretic scenarios (Colonel Blotto, Cournot Duopoly, Stag Hunt, Vickrey Auction, Chicken Game) spanning zero-sum to cooperative dilemmas (Appendix C.2). Experiments are run through OutplayArena², which owns the base game rules, turn sequencing, and scoring.

Selectors. A suite of selectors spanning traditional and adversarial bandit samplers [2, 27], contemporary LLM routers [5, 24], and an LLM self-reflection framework [20] is then incorporated to draw on this portfolio. Adapting selectors from a range of different backgrounds plays an important diagnostic role in understanding the strengths and weaknesses of each method as strategic selectors. Additionally, simple opponent-aware full-information selectors are evaluated post-hoc to measure the selection headroom. We refer the reader to Appendix C.3 to see the implementation details.

Number of Rounds and Seeds. Some strategic benchmarks [26] evaluate models across hundreds of games. To test whether such broad evaluations rest on a consistent signal, we quantify the per-game signal-to-noise ratio (SNR) and find that the evaluation signal is inherently game-specific. A power analysis on single-round fixed-model matchups shows that the share of detectable pairwise comparisons stabilises at $N \approx 150$ seeds in every evaluated game, with Cournot Duopoly separating models nearly twice as cleanly as any other game (Appendix B.3). In our experiments, fixed-model matchups are single-round $T = 1$ with $N = 300$ seeds. Selectors learn online and require repeated play, so we run $T = 300$ rounds per seed with a single seed ($N = 1$) to keep the inference cost tractable. Note that the fielded model’s and fixed opponent’s context is limited to the current round. This ensures that each round is an independent draw of the opponent’s play.

4.2 Results

Strategic Capabilities Vary Across Models and Games. Table 2 reveals that strategic competence is uneven and game dependent. For example, the Llama models are top performers in Cournot Duopoly but fail to make the podium in Chicken Game. In addition, the substantial performance variance among models within individual games highlights an opportunity to adopt a selector. Figure 2a underscores that model scale does not reliably translate into strategic capability. Across our evaluations, we find only a weak positive correlation between model scale and utility (Pearson’s $r = 0.37$), and a moderate negative relationship when accounting for the usage costs ($r = -0.54$). Upon a closer look, we also see that smaller models have game-specific usability. For example in Vickrey Auction, due to its high rationality (i.e. always bidding the true value), the Granite4.1 3B outperforms other much bigger models that range from 24B to 70B parameters. Finally, Figure 2b supports our framing “choosing a model is choosing a strategy”. For each model, we score strategic dispositions from its in-game actions, such as the dare rate in Chicken Game or the produced quantity in Cournot Duopoly. We describe each model by a vector of its seven disposition scores (Table 8), and project these vectors onto their principal components (Appendix D.1). We find that even same family models can differ sharply in disposition. For example, Granite4.1 8B dares in 96% of Chicken Game rounds while its 3B and 30B versions never dare. Families still show a general coherence, as Qwen3.5 and Llama models compete aggressively in Cournot Duopoly, whereas most Granite and Mistral models cooperate. We believe this is an important signal that a strategic selector should learn to infer and exploit.

Strategic Context Needs Special Treatment. We first simplify our selection problem by playing against a fixed unknown opponent, to try and infer strategic learning signals against a stationary fixed

²<https://github.com/OutplayArena/arena>

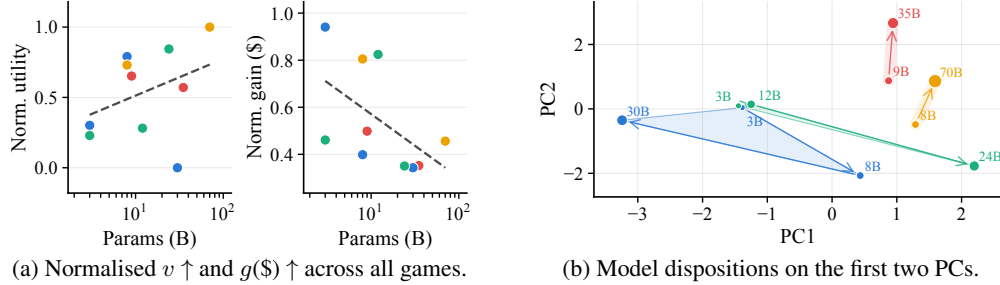


Figure 2: (a) Model scale does not reliably translate into strategic capability. (b) Models differ in strategic disposition. Each point shows a model at the projection of its seven in-game disposition scores onto the first two principal components. **Granite**, **Mistral**, **Llama**, **Qwen3.5**.

Table 2: Normalised in-game utility $v \uparrow$ acquired by each model across 5 games. We report the mean $\pm$ 95% CI, $\times 100$, aggregated across all pairwise matchups, using $T = 1$ round, $N = 300$ seeds per run, and **highlight** the best score for each game.

Model	Chicken Game	Colonel Blotto	Cournot Duopoly	Stag Hunt	Vickrey Auction
Granite4.1 3B	87.5 $\pm$ 0.3	50.6 $\pm$ 0.3	26.0 $\pm$ 1.2	74.3 $\pm$ 2.9	74.3 $\pm$ 0.6
Granite4.1 8B	69.6 $\pm$ 3.0	50.5 $\pm$ 0.3	7.8 $\pm$ 1.5	65.6 $\pm$ 2.2	73.7 $\pm$ 0.6
Granite4.1 30B	87.3 $\pm$ 0.3	50.4 $\pm$ 0.2	13.2 $\pm$ 1.5	72.9 $\pm$ 2.9	69.4 $\pm$ 0.5
Ministral3 3B	86.4 $\pm$ 0.7	48.5 $\pm$ 0.5	18.4 $\pm$ 2.0	73.0 $\pm$ 2.8	72.0 $\pm$ 0.6
Mistral-Nemo 12B	84.2 $\pm$ 1.2	50.3 $\pm$ 0.2	20.0 $\pm$ 2.3	72.7 $\pm$ 2.8	71.9 $\pm$ 0.6
Mistral-Small 3.2 24B	82.2 $\pm$ 1.7	49.7 $\pm$ 0.3	35.2 $\pm$ 2.8	56.1 $\pm$ 1.2	72.5 $\pm$ 0.6
Llama3.1 8B	77.1 $\pm$ 2.3	49.8 $\pm$ 0.4	40.4 $\pm$ 2.7	64.5 $\pm$ 2.0	71.5 $\pm$ 0.7
Llama3.3 70B	70.0 $\pm$ 2.9	50.4 $\pm$ 0.2	48.9 $\pm$ 2.9	75.7 $\pm$ 2.6	73.8 $\pm$ 0.6
Qwen3.5 9B	72.0 $\pm$ 2.8	49.6 $\pm$ 0.4	23.2 $\pm$ 2.3	74.7 $\pm$ 2.6	73.0 $\pm$ 0.6
Qwen3.5 35B	83.7 $\pm$ 1.4	50.3 $\pm$ 0.3	42.0 $\pm$ 2.8	77.2 $\pm$ 2.6	71.4 $\pm$ 0.7

LLM. Table 3 indicates that not all traditional methods transfer equally to the strategic selection setup, with the LLM Selector scoring the lowest overall. Upon closer inspection, we find that it also falls behind nearly every fixed model across most games (Appendix D.3). The Causal LLM Router and RouterDC show relative improvement but are still beaten by several fixed models in multiple games. We attribute this to the fact that strategic settings require special treatment of the game context and the learning signal. Methods that rely on query embeddings for semantic similarity matching (Causal LLM Router, RouterDC), or on LLM reflection (LLM Selector), fail to surface implicit strategic signals. A simple bandit algorithm using Thompson sampling performs best among all selectors in every game, despite relying on nothing more than per-round outcome feedback. However, even in Cournot Duopoly, where the Thompson sampler proves most dominant, it fails to beat the best fixed model (Llama3.3 70B), indicating significant room for improvement (Table 11). We believe a better selector algorithm should learn to directly infer the latent, hidden per-game strategic signal, rather than relying solely on observed game outcomes.

An Interactive Learning Problem. In practice, opponents can also use a selector that adapts its policy each round based on the observed game outcome. To capture this setting, we extend our evaluations and play against adaptive selector opponents. Table 4 reveals that multi-armed bandit algorithms remain dominant. However, the non-stationary adaptations (Sliding-Window TS and Discounted TS) fail to exploit the underlying concept drift in adaptive settings to outclass the plain TS. Similarly, an adversarial variant (EXP3) performs noticeably worse, on par with embedding-based methods. Hence, we note that simple multi-armed bandit adaptations are not sufficient to capture the implicit learning signals shaped by repeated strategic interaction.

5 Lessons Learned

5.1 How to Learn from Strategic Context?

We believe strategic context is multi-faceted and encapsulates the game instructions (i.e. the strategic scenario we are dealing with), the portfolio model dispositions (i.e. understanding our resources to act on this scenario), and the opponent’s actions that shape our scenario. Together, this context is either

Table 3: Normalised net gain $g \uparrow$ (top) and in-game utility $v \uparrow$ (bottom) acquired by each selector, against fixed opponents and across 5 games. We show the mean $\pm$ 95% CI, $\times 100$, averaged across 10 runs (the size of our portfolio), using $T = 300$ rounds, $N = 1$ seed per run, and **highlight** the best score for each game. Selectors’ learning signal was set as their per-round gain $g_{i,t}$.

Selector	Chicken Game	Colonel Blotto	Cournot Duopoly	Stag Hunt	Vickrey Auction
<i>Reports normalised gain $g \uparrow$</i>					
<i>Per-Opponent Full-Information</i>	92.8 $\pm$ 0.4	73.0 $\pm$ 0.2	80.7 $\pm$ 1.5	92.2 $\pm$ 1.0	86.7 $\pm$ 0.5
Thompson Sampler (TS)	85.1 $\pm$ 0.5	71.2 $\pm$ 0.2	47.7 $\pm$ 0.8	76.1 $\pm$ 0.8	74.8 $\pm$ 0.1
RouterDC	84.3 $\pm$ 0.1	68.6 $\pm$ 0.1	36.2 $\pm$ 0.1	74.4 $\pm$ 0.8	72.8 $\pm$ 0.2
Causal LLM Router	78.6 $\pm$ 0.7	63.2 $\pm$ 0.5	31.3 $\pm$ 0.7	74.0 $\pm$ 0.9	64.8 $\pm$ 0.5
LLM Selector	61.0 $\pm$ 0.6	46.9 $\pm$ 0.3	26.9 $\pm$ 0.5	46.6 $\pm$ 1.2	54.4 $\pm$ 0.2
<i>Reports normalised utility $v \uparrow$</i>					
<i>Per-Opponent Full-Information</i>	88.9 $\pm$ 0.7	50.6 $\pm$ 0.3	43.0 $\pm$ 2.6	88.1 $\pm$ 2.1	75.1 $\pm$ 0.7
Thompson Sampler (TS)	87.4 $\pm$ 0.7	50.4 $\pm$ 0.1	39.4 $\pm$ 1.4	78.0 $\pm$ 1.3	73.4 $\pm$ 0.3
RouterDC	87.8 $\pm$ 0.2	50.5 $\pm$ 0.1	19.6 $\pm$ 0.3	76.1 $\pm$ 1.5	72.8 $\pm$ 0.3
Causal LLM Router	79.7 $\pm$ 1.1	50.1 $\pm$ 0.2	27.1 $\pm$ 1.3	72.4 $\pm$ 1.4	72.2 $\pm$ 0.4
LLM Selector	80.6 $\pm$ 1.2	50.1 $\pm$ 0.2	18.1 $\pm$ 1.0	87.0 $\pm$ 2.3	71.5 $\pm$ 0.5

Table 4: Normalised net gain $g \uparrow$, in-game utility $v \uparrow$, and raw usage cost $c \downarrow$ acquired by each selector, against other selectors and in Cournot Duopoly. We report the mean $\pm$ 95% CI, $\times 100$, aggregated across all pairwise matchups, using $T = 300$ rounds, $N = 1$ seed per run, and **highlight** the best score for each metric. Selectors’ learning signal was set as their per-round gain $g_{i,t}$.

Selector	net gain $g \uparrow$	in-game utility $v \uparrow$	usage cost $c (\times 10^{-3} \$) \downarrow$
Thompson Sampler (TS)	72.4 $\pm$ 0.8	47.1 $\pm$ 1.7	0.027 $\pm$ 0.001
Sliding-Window TS	65.9 $\pm$ 0.7	33.2 $\pm$ 1.4	0.017 $\pm$ 0.001
Discounted TS	63.1 $\pm$ 0.8	27.5 $\pm$ 1.6	0.017 $\pm$ 0.001
EXP3	60.2 $\pm$ 0.7	22.1 $\pm$ 1.4	0.020 $\pm$ 0.001
RouterDC	59.6 $\pm$ 0.3	19.3 $\pm$ 0.5	0.002 $\pm$ 0.000
Causal LLM Router	59.1 $\pm$ 0.7	19.7 $\pm$ 1.3	0.019 $\pm$ 0.001
LLM Selector	40.9 $\pm$ 0.7	19.6 $\pm$ 1.3	0.409 $\pm$ 0.004

available as a prior (e.g. through existing game-specific runs across models), or gathered post-hoc by observing the game outcome $\xi_{i,t}$ at the end of each round. Traditional routing and sampling methods fail to make best use of this context. Existing methods either encode the entire context (e.g. game instructions, history) as a whole, as form of vector embedding (RouterDC), without an underlying form or structure, or rely entirely on the observed game outcome (Thompson Sampler) with no regard to the strategic scenario (captured by the game instructions) we are dealing with. A better selector should infer game-specific strategic cues, and map selection choices to their implications for future game states. It should build an understanding of the performance of its own portfolio models, and the opponent under the current scenario.

5.2 Games as Strategic Testbeds

We saw that best-performing models vary noticeably across the 5 evaluated games, each setup capturing a particular strategic scenario (a model-to-model interplay) (Table 2). Moreover, an observation can be made that games vary in difficulty (i.e. complexity). For example, one can say that identifying the equilibrium strategy would require higher model reasoning capabilities in a game like Cournot Duopoly (an economic profit game) than in Colonel Blotto, which relies on a simple per-battlefield resolution rule (the larger number wins). Empirically, this idea is reflected in the correlation between model scale and in-game utility v , which is the highest in Cournot Duopoly (Pearson’s $r = 0.66$), and much lower in games like Stag Hunt ($r = 0.12$) and Colonel Blotto ($r = 0.24$) (Figure 7). Mathematically, there are an infinite number of normal- and extensive-form games that range in complexity and the strategic interplay captured. Future work should therefore establish well-defined methods to classify a game’s difficulty and strategic value. Such methods are

of high value to deployers before they decide where to train and use their agent workflows, as well as towards designing strategic selectors that better understand the underlying strategic learning signal.

5.3 Games as Dynamic Testbeds

Real-world AI deployments are messy. Inputs are noisy, and the agents involved often hold different, sometimes conflicting objectives. Traditional benchmarks fail to capture this, as they test one model at a time against a single fixed correct answer. However, our results show that a model’s competence is shaped by the opponent it faces, and how both sides adapt as the interaction unfolds. This makes game inventories a natural fit for evaluating models in decision-pressured, interactive settings they are actually deployed in practice. Moreover, because there is no single correct answer to memorise, games better resist the training-data contamination that increasingly undermines static benchmarks [22]. Though, we note that such resistance is not absolute. In some games (e.g. Colonel Blotto) models converge toward near-equilibrium play, which can erode the discriminative signal over time even without any contamination.

6 Conclusion and Future Work

We extend game-theoretic evaluations and define strategic selection as a repeated opponent-conditioned meta game layered on top of the base game. Initial evaluations across 5 games and 10 models show that LLM strategic capability varies substantially across games and models, leaving considerable room for a selector to exploit. By adapting a range of selection methods, we report that the existing methods do not transfer equally well even against a simple fixed-model opponent, and that a plain bandit algorithm can surpass structurally heavier pre-trained encoders and LLM reflection methods. Subsequent runs against a selector opponent reveal that simple adaptations like discounted and adversarial bandit algorithms are not able to capture the non-stationary dynamics of opponent-conditioned strategic selection, and fail to outclass the plain sampler. Overall, the results highlight the challenges of strategic model selection, and motivate a selector design that can extract and learn from signals across 3 different dimensions: model capability, usage cost, and strategic disposition, in noise-heavy interactive settings.

Our investigation was scoped to the analysis of 5 games. However, to better understand agent-to-agent interplay and the underlying safety and risk factors, it is important to consider games from a range of fields, including economics, social sciences, and operations research. We believe this choice should reflect both the measurable effect of the game (Appendix B) and the underlying practical application of the selector. We also relied on explicit parsed game actions as a communication method between the models, which underestimates both the strategic capabilities (e.g. persuasion), and the associated risk factors in deployment. A more realistic setting should capture agent-to-agent natural language interactions [25]. We leave such free-form interactions to future work. Finally, high-stakes repeated interactions (e.g. negotiations, auctions) involve a risk factor not captured by expected gain. This can include catastrophic losses, regulatory penalties, or reputational damage. Capturing the variance in realised gain to account for involved risk is another future direction. For example, a risk knob ($\beta \in \mathbb{R}$: $\beta = 0$ is risk-neutral, $\beta > 0$ risk-averse, $\beta < 0$ risk-seeking) could allow users to better align the selector with their deployment policies.

References

- [1] P. Aggarwal, A. Madaan, A. Anand, S. P. Potharaju, S. Mishra, P. Zhou, A. Gupta, D. Rajagopal, K. Kappaganthu, Y. Yang, et al. Automix: Automatically mixing language models. *Advances in Neural Information Processing Systems*, 37:131000–131034, 2024.
- [2] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.
- [3] J. Chen, X. Hu, S. Liu, S. Huang, W.-W. Tu, Z. He, and L. Wen. Llmarena: Assessing capabilities of large language models in dynamic multi-agent environments. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13055–13077, 2024.

- [4] L. Chen, M. Zaharia, and J. Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*, 2023.
- [5] S. Chen, W. Jiang, B. Lin, J. Kwok, and Y. Zhang. Routerdc: Query-based router by dual contrastive learning for assembling large language models. *Advances in Neural Information Processing Systems*, 37:66305–66328, 2024.
- [6] D. Ding, A. Mallick, C. Wang, R. Sim, S. Mukherjee, V. Rühle, L. V. S. Lakshmanan, and A. H. Awadallah. Hybrid LLM: Cost-efficient and quality-aware query routing. In *International Conference on Learning Representations (ICLR)*, 2024.
- [7] J. Duan, R. Zhang, J. Diffenderfer, B. Kailkhura, L. Sun, E. Stengel-Eskin, M. Bansal, T. Chen, and K. Xu. Gtbench: Uncovering the strategic reasoning capabilities of llms via game-theoretic evaluations. *Advances in Neural Information Processing Systems*, 37:28219–28253, 2024.
- [8] D. Heineman, V. Hofmann, I. Magnusson, Y. Gu, N. A. Smith, H. Hajishirzi, K. Lo, and J. Dodge. Signal and noise: A framework for reducing uncertainty in language model evaluation. *arXiv preprint arXiv:2508.13144*, 2025.
- [9] J. J. Horton, A. Filippas, and B. S. Manning. Large language models as simulated economic agents: What can we learn from homo silicus? Technical report, National Bureau of Economic Research, 2023.
- [10] J.-t. Huang, E. J. Li, M. H. Lam, T. Liang, W. Wang, Y. Yuan, W. Jiao, X. Wang, Z. Tu, and M. R. Lyu. How far are we on the decision-making of llms? evaluating llms’ gaming ability in multi-agent environments. *arXiv preprint arXiv:2403.11807*, 2024.
- [11] J. Jia, Z. Yuan, J. Pan, P. McNamara, and D. Chen. Llm strategic reasoning: Agentic study through behavioral game theory. *Advances in Neural Information Processing Systems*, 38: 58318–58350, 2026.
- [12] A. Khatua, H. Zhu, P. Tran, A. Prabhudesai, F. Sadrieh, J. K. Lieberwirth, X. Yu, Y. Fu, M. J. Ryan, J. Pei, et al. Cooperbench: Why coding agents cannot be your teammates yet. *arXiv preprint arXiv:2601.13295*, 2026.
- [13] Q. Mei, Y. Xie, W. Yuan, and M. O. Jackson. A turing test of whether ai chatbots are behaviorally similar to humans. *Proceedings of the National Academy of Sciences*, 121(9):e2313925121, 2024.
- [14] E. Miller. Adding error bars to evals: A statistical approach to language model evaluations. *arXiv preprint arXiv:2411.00640*, 2024.
- [15] Z. Mustahsan, A. Lim, M. Anand, S. Jain, and B. McCann. Stochasticity in agentic evaluations: Quantifying inconsistency with intraclass correlation. *arXiv preprint arXiv:2512.06710*, 2025.
- [16] I. Ong, A. Almahairi, V. Wu, W.-L. Chiang, T. Wu, J. E. Gonzalez, M. W. Kadous, and I. Stoica. RouteLLM: Learning to route LLMs with preference data. In *arXiv preprint arXiv:2406.18665*, 2024.
- [17] C. Qian, Z. Liu, S. Kokane, A. Prabhakar, J. Qiu, H. Chen, Z. Liu, H. Ji, W. Yao, S. Heinecke, et al. xrouter: Training cost-aware llms orchestration system via reinforcement learning. *arXiv preprint arXiv:2510.08439*, 2025.
- [18] V. Raj and S. Kalyani. Taming non-stationary bandits: A bayesian approach. *arXiv preprint arXiv:1707.09727*, 2017.
- [19] N. Raman, T. Lundy, S. Amouyal, Y. Levine, K. Leyton-Brown, and M. Tennenholtz. Steer: Assessing the economic rationality of large language models. *arXiv preprint arXiv:2402.09552*, 2024.
- [20] N. Shinn, F. Cassano, A. Gopinath, K. Narasimhan, and S. Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36: 8634–8652, 2023.

- [21] S. Somerstep, F. M. Polo, A. F. M. de Oliveira, P. Mangal, M. Silva, O. Bhardwaj, M. Yurochkin, and S. Maity. Carrot: A cost aware rate optimal router. *arXiv preprint arXiv:2502.03261*, 2025.
- [22] Y. Sun, H. Wang, D. Li, G. Wang, and H. Zhang. The emperor’s new clothes in benchmarking? a rigorous examination of mitigation strategies for llm benchmark data contamination. *arXiv preprint arXiv:2503.16402*, 2025.
- [23] W. Tang, Y. Zhou, K. Cheng, E. Xu, L. Xiao, and M. Li. Dsgbench: A diverse strategic game benchmark for evaluating llm-based agents in complex decision-making environments. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 16987–16991. IEEE, 2026.
- [24] A. Tsiourvas, W. Sun, and G. Perakis. Causal llm routing: End-to-end regret minimization from observational data. *Advances in Neural Information Processing Systems*, 38:37289–37314, 2026.
- [25] V. C. Vendrell, M. R. Luyten, and M. van der Schaar. Gametalk: Training llms for strategic conversation. *arXiv preprint arXiv:2601.16276*, 2026.
- [26] H. Wang, X. Feng, L. Li, Y. Guo, Z. Qin, D. Sui, and L. Kong. Tmgbench: A systematic game benchmark for evaluating strategic reasoning abilities of llms. *arXiv preprint arXiv:2410.10479*, 2024.
- [27] R. T. William et al. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.
- [28] H. Woisetschlager, A. Mammadli, R. Zhang, and S. Wang. Cost-optimal llm routing with limited user feedback under user satisfaction guarantees. *arXiv preprint arXiv:2606.19376*, 2026.
- [29] H. Woisetschlager, R. Zhang, S. Wang, and H. A. Jacobsen. Mess+: Dynamically learned inference-time llm routing in model zoos with service level guarantees. *Advances in Neural Information Processing Systems*, 38:54034–54067, 2026.
- [30] Y. Wu, X. Tang, T. Mitchell, and Y. Li. Smartplay: A benchmark for llms as intelligent agents. In *International Conference on Learning Representations*, volume 2024, pages 1543–1561, 2024.
- [31] Y. Xie, K. Wang, B. Cheng, J. Yao, Z. Sha, A. Duffy, Y. Xi, H. Mei, C. Tan, C. Wei, et al. Memo: Memory-augmented model context optimization for robust multi-turn multi-agent llm games. *arXiv preprint arXiv:2603.09022*, 2026.
- [32] L. Xu, Z. Hu, D. Zhou, H. Ren, Z. Dong, K. Keutzer, S. K. Ng, and J. Feng. Magic: Investigation of large language model powered multi-agent in cognition, adaptability, rationality and collaboration. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7315–7332, 2024.
- [33] Z. Xu, Z. Xu, X. Yi, H. Yuan, M. Guang, K. Long, X. Chen, Y. Wu, C. Yu, and Y. Wang. Vs-bench: Evaluating vlms for strategic abilities in multi-agent environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21561–21572, 2026.
- [34] X. Yi, Z. Zhou, C. Cao, Q. Niu, T. Liu, and B. Han. From debate to equilibrium: Belief-driven multi-agent llm reasoning via bayesian nash equilibrium. *arXiv preprint arXiv:2506.08292*, 2025.
- [35] H. Yuan, Z. Xu, Z. Tan, X. Yi, M. Guang, K. Long, H. Hui, B. Li, X. Chen, B. Zhao, et al. Marshal: Incentivizing multi-agent reasoning via self-play with strategic llms. In *International Conference on Learning Representations*, volume 2026, pages 6209–6236, 2026.

Appendix

A	Glossary of Notation	12
B	More Details about Noise in Strategic Play	12
B.1	Signal and Noise Ratio	12
B.2	Number of Seeds	12
B.3	Analysis	13
C	More Details about Experimental Setup	14
C.1	Models	14
C.2	Games	14
C.3	Selectors	15
D	More Results	16
D.1	Fixed Model Disposition Scores	16
D.2	Fixed Model vs. Fixed Model	17
D.3	Selector vs. Fixed Model	18

A Glossary of Notation

Table 5: Glossary of notation.

Symbol	Meaning
Meta Game Γ (LLM selection game)	
$i, -i$	player indices, where $-i$ denotes the opponent of i
$t = 1, \dots, T$	round index of the game
N	number of seeds in the game (independent replications)
$S_i = \{1, \dots, M_i\}$	finite set of pure strategies, i.e. player i 's model portfolio
$\sigma_i \in \Delta(S_i)$	mixed strategy for player i , a probability simplex over S_i
$x_{i,t} \in \mathcal{X}$	context observed by player i before round t
$\xi_{i,t}$	game outcome produced by player i on round t
$s_{i,t} \sim \sigma_i$	pure strategy (model choice) fielded by player i at round t
$v_{i,t}$	utility obtained by player i from the base-game outcome
$c_{i,t}$	inference cost incurred by player i (tok/s or tok/\$)
$g_{i,t}$	net gain that captures the utility-cost trade-off
$\theta_{i,t}$	internal state of player i 's selector at round t
Base Game $\mathcal{G}$ (e.g. Colonel Blotto, Cournot Duopoly)	

B More Details about Noise in Strategic Play

B.1 Signal and Noise Ratio

Following Heineman et al. [8], we first define a descriptive method that characterises signal and noise levels across games. Fix a game $\mathcal{G}$ and let S denote its set of models under evaluation, with $j, k \in S$ ranging over the pair of models compared in $\mathcal{G}$. For a model $s \in S$, let $\tau = 1, \dots, N$ index its seeds, where a seed is a single independent evaluation run. Let $v_{\mathcal{G},s,\tau}$ be the bounds-normalised score on seed τ of $\mathcal{G}$ played by model s . Let $\bar{v}_{\mathcal{G},s}$ denote the mean of $v_{\mathcal{G},s,\tau}$ over all N seeds, and let $\bar{v}_{\mathcal{G}}$ denote the mean of $\bar{v}_{\mathcal{G},s}$ over all models $s \in S$. Noise, signal, and signal-to-noise ratio (SNR) are then

$$\text{noise}(\mathcal{G}, s) = \frac{\text{Std}_{\tau}(v_{\mathcal{G},s,\tau})}{\bar{v}_{\mathcal{G},s}}, \quad \text{signal}(\mathcal{G}) = \frac{\max_{j,k \in S} |\bar{v}_{\mathcal{G},j} - \bar{v}_{\mathcal{G},k}|}{\bar{v}_{\mathcal{G}}}, \quad \text{SNR}(\mathcal{G}) = \frac{\text{signal}(\mathcal{G})}{\text{noise}(\mathcal{G})}. \quad (3)$$

Here $\text{Std}_{\tau}(\cdot)$ denotes the standard deviation taken over the seed index τ , and $\overline{\text{noise}(\mathcal{G})}$ denotes $\text{noise}(\mathcal{G}, s)$ averaged over every model $s \in S$. Noise is a coefficient of variation, a model's own seed-to-seed spread relative to its mean. Signal is the largest gap between any two models' means, relative to the game's overall mean. SNR is high when models are well separated relative to how much a single model's score jitters across seeds, and low when the two quantities are comparable.

B.2 Number of Seeds

Given the SNR, we want to determine the minimum number of seeds N that lets us report statistically significant outcomes. This is a trade-off between cost (the number of model inferences we run) and reliability (our statistical confidence in the reported results).

Decision accuracy. Heineman et al. [8] suggests calculating a decision accuracy metric to determine the number of seeds N needed to resolve the pairwise gaps between models. Let $\mathcal{P}$ denote the set of model pairs compared in $\mathcal{G}$, let n denote a candidate (small) batch size, and let $N \gg n$ denote a large reference batch size. Let $\bar{v}_{\mathcal{G},s}^{(n)}$ denote model s 's mean bounds-normalised score over n seeds. Decision accuracy at batch size n is then

$$\text{DA}(n) = \frac{1}{|\mathcal{P}|} \sum_{\{s_A, s_B\} \in \mathcal{P}} \mathbb{1} \left[\text{sign} \left(\bar{v}_{\mathcal{G},s_A}^{(n)} - \bar{v}_{\mathcal{G},s_B}^{(n)} \right) = \text{sign} \left(\bar{v}_{\mathcal{G},s_A}^{(N)} - \bar{v}_{\mathcal{G},s_B}^{(N)} \right) \right], \quad (4)$$

the fraction of the pairs in $\mathcal{P}$ whose sign at batch size n agrees with the sign at the reference batch size N . Every n below N is estimated as the mean over repeated independent subsamples of the N -seed reference, drawn per model pair so each draw mirrors an actual N -seed experiment run at batch size n . However, decision accuracy needs a reference batch, and in game-theoretic evaluations this reference is often not remotely the scale of traditional benchmarks, since the game-theoretic reference batch is self-generated instead of drawn from an external test suite.

Minimum detectable effect. A more practical approach is to use the minimum detectable effect (MDE), which does not need a reference batch, and asks directly whether the current N and opponent-pool size is sufficient to resolve the pairwise gaps we report. Power analysis of Miller [14] (Eqs. 9-10), for two models s_A and s_B with M shared opponents and K seeds played against each opponent, gives the MDE as

$$\delta(K) = (z_{\alpha/2} + z_{\beta}) \sqrt{\frac{\omega^2 + \sigma_{s_A}^2/K + \sigma_{s_B}^2/K}{M}}, \quad (5)$$

$$\omega^2 = \text{Var}_o(\bar{v}_{\mathcal{G},s_A,o} - \bar{v}_{\mathcal{G},s_B,o}), \quad \sigma_{s_A}^2 = \mathbb{E}_o[\text{Var}_t(v_{\mathcal{G},s_A,o,t})], \quad \sigma_{s_B}^2 = \mathbb{E}_o[\text{Var}_t(v_{\mathcal{G},s_B,o,t})]. \quad (6)$$

Here o ranges over the M shared opponents ($M = 8$ in a 10-model pool experiment), and $t = 1, \dots, K$ indexes seeds played against a given opponent. z_{γ} denotes the $(1 - \gamma)$ quantile of the standard normal distribution, so $\alpha = 0.05$ (giving $z_{\alpha/2}$) and power $1 - \beta = 0.8$ (giving z_{β}) are Miller [14]’s own defaults for significance and power. ω^2 is the variance, across the M shared opponents, of the cluster-mean gap between s_A and s_B , i.e. how much that gap swings depending on the opponent, and $\sigma_{s_A}^2$ and $\sigma_{s_B}^2$ are each model’s mean within-opponent variance across seeds. All three quantities are estimated from a reference (e.g. $N = 300$). $\delta(K)$ is the smallest true skill gap detectable at that K , for a fixed opponent-pool size. For game $\mathcal{G}$ evaluated on a model pool S , the detectable fraction at K seeds is

$$\rho_{\mathcal{G}}(K) = \frac{1}{\binom{|S|}{2}} \sum_{\{s_A, s_B\} \subset S} \mathbb{1}[|\bar{v}_{\mathcal{G},s_A} - \bar{v}_{\mathcal{G},s_B}| \geq \delta(K)], \quad (7)$$

the share of the $\binom{|S|}{2}$ pairwise gaps in $\mathcal{G}$ that exceed $\delta(K)$. Sweeping K per game produces a rough signal on the statistical significance of pairwise model comparisons from that game.

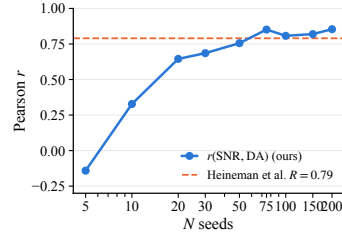
B.3 Analysis

Scope of the analysis. All signal, noise, and seed-stabilisation analyses in this section are computed on fixed-model matchups, using single-round episodes ($T = 1$). Each seed is therefore an independent draw of the base game, and the estimates characterise the inherent variability of strategic play between two given models. We note that our analyses does not directly transfer to selector runs, where selector’s own choices can evolve over the rounds.

Signal-to-noise ratio is not game-agnostic. Cournot Duopoly separates models almost twice as cleanly as any other game, while Colonel Blotto carries the lowest signal (Figure 3a). A qualitative investigation shows that this stems from all models playing Colonel Blotto very similarly, all converging to an equilibrium of an even distribution across battlefields. This SNR ranking validity is confirmed independently by decision accuracy (Figure 3b). Holding the seed count n fixed and correlating SNR against decision accuracy across the five games gives Pearson $r \approx 0.81$ to 0.85 for $n \geq 75$, close to Heineman et al. [8]’s reported $R = 0.79$, indicating that our SNR metric captures the same effect as theirs.

Game	Signal	Noise	SNR
Cournot Duopoly	1.431	1.537	0.931
Chicken Game	0.224	0.357	0.627
Stag Hunt	0.315	0.559	0.564
Vickrey Auction	0.066	0.129	0.512
Colonel Blotto	0.040	0.089	0.449

(a)



(b)

Figure 3: Statistical analysis of signal and noise across 5 games and 4 model families. (a) Signal, noise, and SNR per game, $n = 300$. (b) Similar to the original paper ($R = 0.79$) we show that our correlation holds at $r \approx 0.81$ from $n = 75$ onwards.

Decision accuracy results in Figure 4b follow the same ranking. Cournot Duopoly reaches 95% decision accuracy at the smallest seed count of the five games, $n = 20$, while Colonel Blotto needs the largest, $n = 200$, consistent with its near-zero signal. No single seed count clears 95% decision accuracy on every game before $n = 200$.

In the power analysis (Figure 4a), some games never clear the significance bar even at the full seed budget, while Cournot Duopoly has the highest ratio of detectable pairwise comparisons, confirming the ranking from both the SNR and the decision accuracy analysis. The detection rate converges after around ≥ 150 seeds for the games that do clear the bar. For the primary investigation, we fix the number of seeds to $N = 300$.

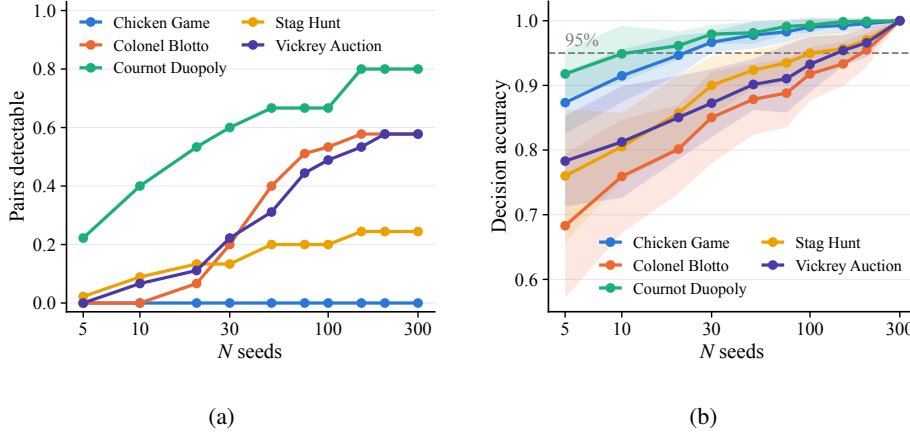


Figure 4: Statistical analysis of signal and noise across 5 games and 4 model families. (a) Fraction of all 45 reported pairwise gaps that are statistically detectable. (b) Decision accuracy, mean over 30 subsample draws per n .

C More Details about Experimental Setup

C.1 Models

Our experimental model portfolio spans 10 open-weight models with three size tiers, listed in Table 6. Every model is served locally through Ollama. Every model call uses a sampling temperature of 1.0 and nucleus sampling (top-p) of 0.9. Every experiment runs on a workstation with 64 physical CPU cores (128 logical cores) and two NVIDIA RTX PRO 6000 Blackwell Workstation Edition GPUs, under Linux 7.0.0-28-generic (x86-64, glibc 2.39) with Python 3.12.3.

Table 6: Model portfolio used across experiments.

Model	Hugging Face repository
granite4.1:3b	ibm-granite/granite-4.1-3b
granite4.1:8b	ibm-granite/granite-4.1-8b
granite4.1:30b	ibm-granite/granite-4.1-30b
llama3.1:8b-instruct-q4_K_M	meta-llama/Llama-3.1-8B-Instruct
llama3.3:70b	meta-llama/Llama-3.3-70B-Instruct
ministral-3:3b	mistralai/Ministral-3-3B-Instruct-2512
mistral-nemo:12b	mistralai/Mistral-Nemo-Instruct-2407
mistral-small13.2:24b	mistralai/Mistral-Small-3.2-24B-Instruct-2506
qwen3.5:9b	Qwen/Qwen3.5-9B
qwen3.5:35b	Qwen/Qwen3.5-35B-A3B

C.2 Games

The base game $\mathcal{G}$ is instantiated as one of five two-player games. For each game we have kept the default configurations provided by OutplayArena. We provide brief details about each game, and refer the reader to the source outlined in Table 7 for further information.

- **Colonel Blotto** is a zero-sum game where two players split a fixed budget across several battlefields. It measures how a model reallocates its strategy in response to an opponent’s shifting distribution.

- **Cournot Duopoly** is a simultaneous quantity-setting game where two firms choose outputs that jointly fix market price and profit. It isolates a single continuous best-response signal that separates models by how closely each converges toward the equilibrium quantity.
- **Stag Hunt** is a coordination game with a risk-dominant and a payoff-dominant equilibrium. It measures whether a model favors the safer but lower-payoff option over the cooperative one when the opponent’s action is uncertain.
- **Vickrey Auction** is a second-price sealed-bid auction where each bidder submits a value-based bid. It measures how closely a model’s bid tracks its private valuation under the auction’s truthful-bidding equilibrium.
- **Chicken Game** is an anti-coordination game built around brinkmanship, where mutual insistence is worst for both players. It measures a model’s willingness to escalate or concede as the risk of mutual loss grows.

Table 7: The five base games $\mathcal{G}$ used across experiments, all played between 2 players for 1 round per episode.

Game	Structure	Source
Chicken Game	anti-coordination, brinkmanship	https://en.wikipedia.org/wiki/Chicken_(game)
Colonel Blotto	zero-sum resource allocation	https://en.wikipedia.org/wiki/Blotto_game
Cournot Duopoly	simultaneous quantity competition	https://en.wikipedia.org/wiki/Cournot_competition
Stag Hunt	coordination, risk dominance	https://en.wikipedia.org/wiki/Stag_hunt
Vickrey Auction	second-price sealed-bid auction	https://en.wikipedia.org/wiki/Vickrey_auction

Table 8: Strategic disposition metrics inferred from in-game play for each game. All metrics and averaged across rounds and lie in $[0, 1]$.

Game	Disposition	Definition
Chicken Game	Risk appetite	Fraction of rounds choosing dare over swerve.
Cournot Duopoly	Cooperativeness	Collusion index; 1 = joint-profit quantity, 0 = at or beyond Nash.
	Aggressiveness	Own minus opponent quantity, as a share of the maximum quantity.
Stag Hunt	Risk appetite	Fraction of rounds choosing stag over the safe hare.
	Cooperativeness	Fraction of rounds landing on mutual stag.
Vickrey Auction	Risk appetite	Mean own bid as a share of the maximum bid.
	Rationality	One minus the mean $ \text{bid} - \text{value} $; 1 = always bids its true value.

C.3 Selectors

We provide the implementation detail of each selector.

Fixed Model. One fixed model across all seeds N that keeps context and past outcomes across rounds $t = 1, \dots, T$ unless otherwise stated.

Per-Opponent Full-Information. Given the identity of the fixed opponent, it looks up the round-robin table (our prior) once at construction and commits to whichever candidate scores highest against that opponent, fielding it every round with no further adaptation.

LLM Selector. Uses llama3.3:70b as the selecting model and relies on the Reflexion framework [20] to pick the model to field. Namely, each round it is queried once with a description of the game, an entry per portfolio model (tier and per-token price), and a log past rounds covering both its own and the opponent’s model, action, and signal. It parses a JSON response giving the picked model, a confidence, and a reasoning. Prior information enters only as in-context history, a rolling log of past rounds inside the prompt, with default sampling temperature = 1.0.

RouterDC. Adapted from [5]. Each round a frozen sentence encoder embeds the current query, built from this player’s own and the opponent’s model, action, and signal history, and the pick is whichever candidate’s trained embedding has highest cosine similarity to that query embedding. The embeddings come from a checkpoint fit offline, once per game. All prior information is folded into this pre-trained checkpoint at training time.

Causal LLM Router. We follow the doubly robust regret-minimising rule of Tsiourvas et al. [24]. Each round it draws a candidate from a softmax over its current per-(candidate, opponent) belief, with a floor that keeps every candidate at nonzero pick probability. It then folds in a doubly robust correction of the observed signal, using the pick probability it assigned as its own propensity instead of fitting a separate propensity model. We use the defaults `surrogate = softmax`, `tau = 1.0`, and `propensity_floor = 0.05`.

Thompson Sampler. It keeps a per-model Gaussian belief over the chosen signal that is refined as a running mean after every round. Each round it draws one sample per model from its current belief and fields the model with the highest draw. Additionally, we implement two non-stationary variants that change the belief update. The discounted variant (*Dis. Thompson Sampler*) multiplies every model’s accumulated pseudo-count by the discount factor before folding in the new observation, so older evidence decays exponentially. The sliding-window variant (*SW Thompson Sampler*) keeps only the W (denoting the window size) most recent observations, and drops older ones from the belief entirely. We use `discount = 0.99` in the discounted variant and `window = 50` in the sliding-window variant.

EXP3. An exponential-weights bandit updated by an importance-weighted multiplicative rule on the observed signal. Its mixed strategy blends the weight-proportional distribution with a uniform exploration floor, so every candidate keeps nonzero pick probability, with default `exploration_rate = 0.1`.

D More Results

This section reports the full pairwise results that the main text summarises. Section D.1 captures the complete in-game disposition analysis inferred from fixed model matchups. Section D.2 gives the complete fixed-model-versus-fixed-model matchup results across all 5 games. Section D.3 then gives the full selector-versus-fixed-model matchup results.

D.1 Fixed Model Disposition Scores

Table 9: We standardize each disposition score across models and project the resulting vectors onto their two directions of largest variation, which together capture 69% of the differences between models. Our interpretation is that PC1 measures how competitive a model is, while PC2 signals trust in the opponent. Bold marks loadings with a magnitude of at least 0.35.

Game	Disposition	PC1	PC2
Chicken Game	Risk appetite	+0.35	−0.12
Cournot Duopoly	Cooperativeness	− 0.45	− 0.41
	Aggressiveness	+0.49	+0.35
Stag Hunt	Risk appetite	− 0.38	+0.50
	Cooperativeness	−0.34	+0.54
Vickrey Auction	Risk appetite	+0.38	+0.29
	Rationality	+0.19	−0.28
Explained variance		43%	26%

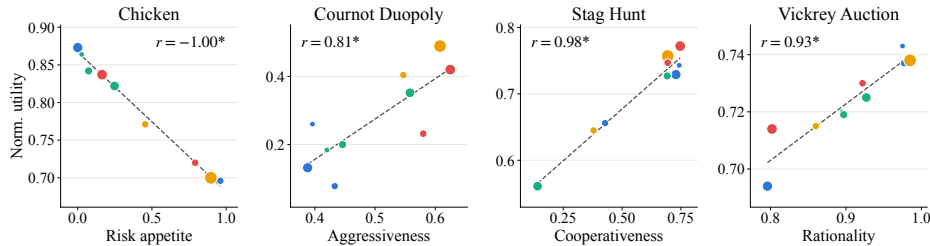


Figure 5: Normalised in-game utility $v \uparrow$ against the disposition expected to drive each game outcome. Dashed lines show the linear fit and r its Pearson correlation. **Granite**, **Mistral**, **Llama**, **Qwen3.5**.

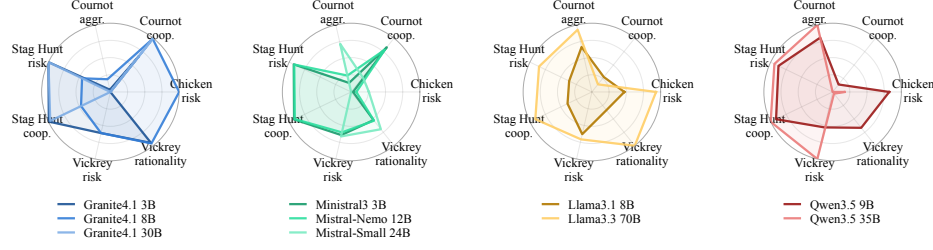


Figure 6: Strategic dispositions inferred from in-game behavior for each model family. Each axis is min-max normalised across the 10 models of the portfolio. **Granite**, **Mistral**, **Llama**, **Qwen3.5**.

D.2 Fixed Model vs. Fixed Model

Table 10: Net gain, normalised utility, and normalised cost acquired by each model across 5 games. We report the mean $\pm$ 95% CI, $\times 100$, aggregated across all pairwise matchups, using $T = 1$ round, $N = 300$ seeds per run. **Highlight** marks the best score per game within each metric block.

Model	Chicken Game	Colonel Blotto	Cournot Duopoly	Stag Hunt	Vickrey Auction
<i>Reports normalised gain $g \uparrow$</i>					
Granite4.1 3B	93.8 $\pm$ 0.1	75.2 $\pm$ 0.1	63.0 $\pm$ 0.6	87.2 $\pm$ 1.4	87.2 $\pm$ 0.3
Granite4.1 8B	82.3 $\pm$ 1.5	74.5 $\pm$ 0.1	46.9 $\pm$ 0.8	72.0 $\pm$ 1.2	85.8 $\pm$ 0.3
Granite4.1 30B	87.0 $\pm$ 0.1	74.0 $\pm$ 0.1	48.6 $\pm$ 0.9	78.1 $\pm$ 1.4	80.1 $\pm$ 0.3
Ministral3 3B	86.5 $\pm$ 0.4	73.4 $\pm$ 0.2	53.5 $\pm$ 1.0	78.2 $\pm$ 1.4	84.2 $\pm$ 0.3
Mistral-Nemo 12B	92.1 $\pm$ 0.6	75.0 $\pm$ 0.1	59.4 $\pm$ 1.2	86.2 $\pm$ 1.4	85.8 $\pm$ 0.3
Mistral-Small 3.2 24B	77.4 $\pm$ 0.9	74.2 $\pm$ 0.1	58.8 $\pm$ 1.4	69.9 $\pm$ 0.7	84.2 $\pm$ 0.3
Llama3.1 8B	88.4 $\pm$ 1.1	74.8 $\pm$ 0.2	69.8 $\pm$ 1.4	82.0 $\pm$ 1.0	85.6 $\pm$ 0.3
Llama3.3 70B	78.2 $\pm$ 1.4	73.2 $\pm$ 0.1	65.7 $\pm$ 1.5	79.3 $\pm$ 1.3	83.7 $\pm$ 0.3
Qwen3.5 9B	79.2 $\pm$ 1.4	74.0 $\pm$ 0.2	57.9 $\pm$ 1.1	78.8 $\pm$ 1.3	85.5 $\pm$ 0.4
Qwen3.5 35B	80.5 $\pm$ 0.7	72.2 $\pm$ 0.2	64.4 $\pm$ 1.4	74.7 $\pm$ 1.3	83.8 $\pm$ 0.4
<i>Reports normalised utility $v \uparrow$</i>					
Granite4.1 3B	87.5 $\pm$ 0.3	50.6 $\pm$ 0.3	26.0 $\pm$ 1.2	74.3 $\pm$ 2.9	74.3 $\pm$ 0.6
Granite4.1 8B	69.6 $\pm$ 3.0	50.5 $\pm$ 0.3	7.8 $\pm$ 1.5	65.6 $\pm$ 2.2	73.7 $\pm$ 0.6
Granite4.1 30B	87.3 $\pm$ 0.3	50.4 $\pm$ 0.2	13.2 $\pm$ 1.5	72.9 $\pm$ 2.9	69.4 $\pm$ 0.5
Ministral3 3B	86.4 $\pm$ 0.7	48.5 $\pm$ 0.5	18.4 $\pm$ 2.0	73.0 $\pm$ 2.8	72.0 $\pm$ 0.6
Mistral-Nemo 12B	84.2 $\pm$ 1.2	50.3 $\pm$ 0.2	20.0 $\pm$ 2.3	72.7 $\pm$ 2.8	71.9 $\pm$ 0.6
Mistral-Small 3.2 24B	82.2 $\pm$ 1.7	49.7 $\pm$ 0.3	35.2 $\pm$ 2.8	56.1 $\pm$ 1.2	72.5 $\pm$ 0.6
Llama3.1 8B	77.1 $\pm$ 2.3	49.8 $\pm$ 0.4	40.4 $\pm$ 2.7	64.5 $\pm$ 2.0	71.5 $\pm$ 0.7
Llama3.3 70B	69.9 $\pm$ 2.9	50.4 $\pm$ 0.2	48.9 $\pm$ 2.9	75.7 $\pm$ 2.6	73.8 $\pm$ 0.6
Qwen3.5 9B	72.0 $\pm$ 2.8	49.6 $\pm$ 0.4	23.2 $\pm$ 2.3	74.7 $\pm$ 2.6	73.0 $\pm$ 0.6
Qwen3.5 35B	83.7 $\pm$ 1.4	50.3 $\pm$ 0.3	42.0 $\pm$ 2.8	77.2 $\pm$ 2.6	71.4 $\pm$ 0.7
<i>Reports normalised cost $c \downarrow$</i>					
Granite4.1 3B	0.0 $\pm$ 0.0	0.1 $\pm$ 0.0	0.0 $\pm$ 0.0	0.0 $\pm$ 0.0	0.0 $\pm$ 0.0
Granite4.1 8B	5.0 $\pm$ 0.0	1.5 $\pm$ 0.0	13.9 $\pm$ 0.3	21.6 $\pm$ 1.1	2.0 $\pm$ 0.1
Granite4.1 30B	13.3 $\pm$ 0.0	2.4 $\pm$ 0.1	16.1 $\pm$ 0.9	16.7 $\pm$ 0.0	9.3 $\pm$ 0.4
Ministral3 3B	13.4 $\pm$ 0.2	1.7 $\pm$ 0.0	11.4 $\pm$ 0.2	16.6 $\pm$ 0.2	3.6 $\pm$ 0.1
Mistral-Nemo 12B	0.1 $\pm$ 0.0	0.2 $\pm$ 0.0	1.2 $\pm$ 0.1	0.4 $\pm$ 0.1	0.3 $\pm$ 0.0
Mistral-Small 3.2 24B	27.4 $\pm$ 0.8	1.3 $\pm$ 0.0	17.7 $\pm$ 0.7	16.3 $\pm$ 0.8	4.2 $\pm$ 0.1
Llama3.1 8B	0.2 $\pm$ 0.0	0.1 $\pm$ 0.0	0.9 $\pm$ 0.0	0.4 $\pm$ 0.0	0.3 $\pm$ 0.0
Llama3.3 70B	13.6 $\pm$ 0.0	3.9 $\pm$ 0.1	17.5 $\pm$ 0.6	17.0 $\pm$ 0.0	6.4 $\pm$ 0.2
Qwen3.5 9B	13.6 $\pm$ 0.0	1.5 $\pm$ 0.0	7.4 $\pm$ 0.0	17.2 $\pm$ 0.0	2.0 $\pm$ 0.0
Qwen3.5 35B	22.8 $\pm$ 0.0	6.0 $\pm$ 0.3	13.2 $\pm$ 0.0	27.8 $\pm$ 0.0	3.7 $\pm$ 0.2

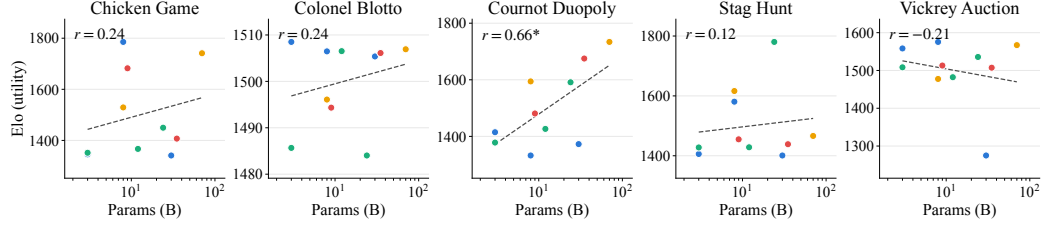


Figure 7: Model elo score (bootstrapped) vs. scale. **Granite**, **Mistral**, **Llama**, **Qwen3.5**.

D.3 Selector vs. Fixed Model

Table 11: Normalised net gain $g \uparrow$ (top) and in-game utility $v \uparrow$ (bottom) acquired by each selector and fixed model from full pairwise matchups against each other, and across 5 games. We show the mean $\pm$ 95% CI, $\times 100$, using $T = 300$ rounds, $N = 1$ seed per run, and **highlight** the best score for each game. Selectors’ learning signal was set as their per-round gain $g_{i,t}$.

Method	Chicken Game	Colonel Blotto	Cournot Duopoly	Stag Hunt	Vickrey Auction
<i>Reports normalised gain $g \uparrow$</i>					
<i>Per-Opponent Full-Information</i>	92.8 ± 0.4	73.0 ± 0.2	80.7 ± 1.5	92.2 ± 1.0	86.7 ± 0.5
Granite4.1 3B	89.7 ± 0.1	71.6 ± 0.1	64.0 ± 0.5	89.2 ± 0.8	83.4 ± 0.2
Granite4.1 8B	84.5 ± 0.8	70.5 ± 0.1	49.8 ± 0.5	81.1 ± 0.7	82.0 ± 0.2
Granite4.1 30B	77.3 ± 0.1	59.1 ± 0.1	40.9 ± 0.5	76.6 ± 0.8	68.6 ± 0.2
Ministral3 3B	86.5 ± 0.2	67.3 ± 0.2	53.3 ± 0.7	83.6 ± 0.8	78.9 ± 0.2
Mistral-Nemo 12B	91.9 ± 0.4	75.1 ± 0.1	61.0 ± 0.8	90.9 ± 0.8	85.8 ± 0.2
Mistral-Small 3.2 24B	80.7 ± 0.6	64.2 ± 0.1	57.2 ± 0.9	66.6 ± 0.4	75.7 ± 0.2
Llama3.1 8B	90.4 ± 0.7	74.6 ± 0.1	72.6 ± 0.9	81.9 ± 0.7	85.6 ± 0.3
Llama3.3 70B	71.9 ± 0.9	58.2 ± 0.1	61.3 ± 0.9	73.8 ± 0.8	69.8 ± 0.2
Qwen3.5 9B	80.4 ± 0.8	65.7 ± 0.1	60.3 ± 0.8	81.2 ± 0.8	77.0 ± 0.2
Qwen3.5 35B	42.5 ± 0.5	25.1 ± 0.1	25.1 ± 0.9	42.4 ± 0.8	35.6 ± 0.3
Thompson Sampler (TS)	85.1 ± 0.5	71.2 ± 0.2	47.7 ± 0.8	76.1 ± 0.8	74.8 ± 0.1
RouterDC	84.3 ± 0.1	68.6 ± 0.1	36.2 ± 0.1	74.4 ± 0.8	72.8 ± 0.2
Causal LLM Router	78.6 ± 0.7	63.2 ± 0.5	31.3 ± 0.7	74.0 ± 0.9	64.8 ± 0.5
LLM Selector	61.0 ± 0.6	46.9 ± 0.3	26.9 ± 0.5	46.6 ± 1.2	54.4 ± 0.2
<i>Reports normalised utility $v \uparrow$</i>					
<i>Per-Opponent Full-Information</i>	88.9 ± 0.7	50.6 ± 0.3	43.0 ± 2.6	88.1 ± 2.1	75.1 ± 0.7
Granite4.1 3B	86.7 ± 0.2	50.5 ± 0.1	35.3 ± 1.0	85.6 ± 1.5	74.1 ± 0.5
Granite4.1 8B	78.2 ± 1.6	50.3 ± 0.1	8.8 ± 1.0	71.3 ± 1.4	73.2 ± 0.5
Granite4.1 30B	86.7 ± 0.2	50.3 ± 0.1	13.9 ± 1.0	85.4 ± 1.5	69.4 ± 0.4
Ministral3 3B	86.8 ± 0.5	48.5 ± 0.3	20.3 ± 1.4	81.1 ± 1.6	71.6 ± 0.5
Mistral-Nemo 12B	83.7 ± 0.8	50.2 ± 0.2	22.1 ± 1.5	81.7 ± 1.7	71.6 ± 0.5
Mistral-Small 3.2 24B	82.0 ± 1.2	49.0 ± 0.2	35.0 ± 1.8	53.9 ± 0.8	72.1 ± 0.5
Llama3.1 8B	80.8 ± 1.4	49.3 ± 0.3	45.2 ± 1.7	63.8 ± 1.4	71.2 ± 0.5
Llama3.3 70B	77.7 ± 1.8	50.4 ± 0.1	56.5 ± 1.8	81.5 ± 1.6	73.6 ± 0.5
Qwen3.5 9B	79.1 ± 1.6	49.8 ± 0.2	39.0 ± 1.7	80.7 ± 1.6	72.4 ± 0.5
Qwen3.5 35B	85.0 ± 0.9	50.2 ± 0.2	50.2 ± 1.8	84.8 ± 1.6	71.2 ± 0.6
Thompson Sampler (TS)	87.4 ± 0.7	50.4 ± 0.1	39.4 ± 1.4	78.0 ± 1.3	73.4 ± 0.3
RouterDC	87.8 ± 0.2	50.5 ± 0.1	19.6 ± 0.3	76.1 ± 1.5	72.8 ± 0.3
Causal LLM Router	79.7 ± 1.1	50.1 ± 0.2	27.1 ± 1.3	72.4 ± 1.4	72.2 ± 0.4
LLM Selector	80.6 ± 1.2	50.1 ± 0.2	18.1 ± 1.0	87.0 ± 2.3	71.5 ± 0.5