Can LLMs Play the Game They Talk? A Linguistic and Behavioral Analysis of Rationality in Kuhn Poker

Presented by **Szu-Chi Lin**

Graduate Institute of Linguistics

National Taiwan University

紙上談兵還是策略玩家？
語言模型於 **Kuhn** 撲克中
理性行為的語言與行動分析

The Architecture of Choice, Without the Mind of Agency: *Do Transformer-Based LLMs (GPT-4o, GPT-4-Turbo, o3-mini) Exhibit Strategic Rationality in Kuhn Poker?*

What is "*Rationality*" (Operationalized)?

From **classical game theory**:

- Belief Updating
- Expected Value (EV) Calculation
- In-Game Adaptation

Multidisciplinary Framework

- **Machine Learning**: LLM architecture, autoregressiveness, prompt-conditioning, XAI
- **Economics + Math**: Nash equilibrium, expected utility
- **Linguistics**: Natural language framing, declarative vs procedural
- **Psychology and Cognition**: Cognitive bias, recursive reasoning

Experimental Setting

Testbed: **Kuhn Poker**

- Simple but non-trivial structure
- Fully solved optimal strategies
- Designed to probe decision-making under uncertainty

Game Theory in 1 Minute 🤯
What LLMs Are *Supposed* to “Know”

Strategy

- Pure/mixed
- Best response: A strategy that yields the highest expected payoff given all other players' strategies

Nash equilibrium (NE)

- A strategy profile where no player can gain by unilaterally deviating

NE Indifference Principle

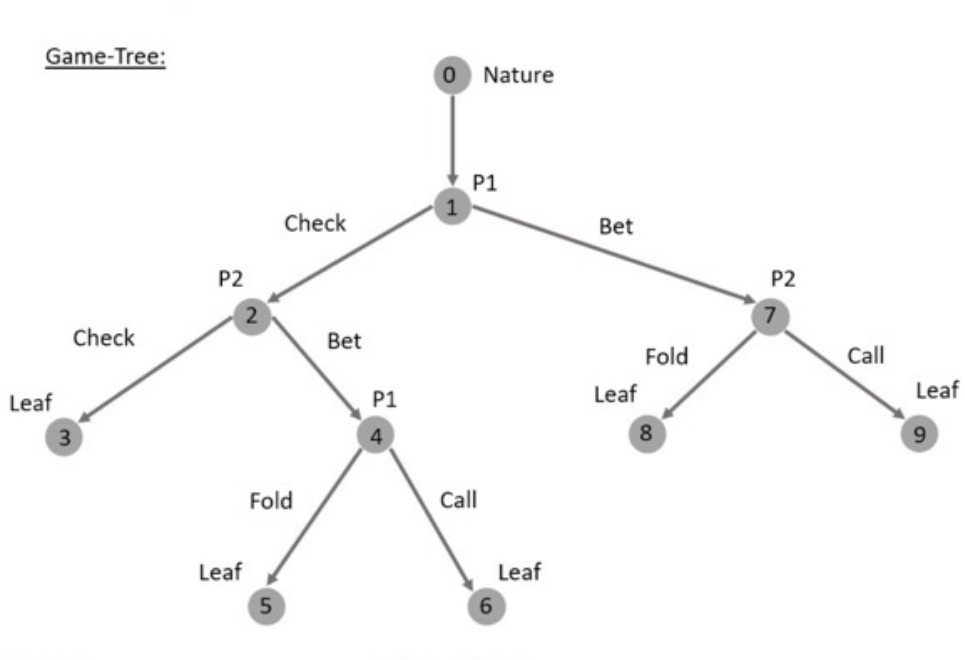
- In equilibrium, all pure strategies in the support of a player's mixed strategy yield identical expected payoffs

Dominance

- (Strictly) dominant Strategy: (Strictly) better than all other strategies regardless of what others do
- A rational player should always play a strictly dominant strategy if it exists, e.g., always call with K, always fold with J

Zero-Sum Game

- One player's gain = the other player's loss
- Kuhn Poker is zero-sum ⇒ cleaner analysis 🤓



Imperfect Information

- A player can't distinguish between game states
- Kuhn Poker random card deals (chance node), i.e., *uncertainty*

Exploitability

- A strategy is exploitable if your opponent can earn more than the NE payoff
- In a *zero-sum game*, if you play Nash, you're *guaranteed* the NE payoff

Rationality

= Choosing the best response under uncertainty, given what you know and believe.

LLMs don’t just need to talk the game—they need to **play** it.

Standing on Their Shoulders (Then Climbing Higher): The Literature and their Limits

Parts, Patterns, But a Missing Whole (Fan et al., 2024)

Framework

- Rationality by parts = utility + belief + optimal action

Findings

- Utility: “corpus-driven” preferences ⇒ “distributional bias”, not “preference modeling”? (Later)
- Belief: Simple patterns in RPS ✅; dynamic inference ❌
- Optimal Action: Implicit Belief → Explicit Belief → Given Belief

Critique

- *Assigned* preferences; non-Bayesian belief update
- Sidesteps **language** modeling strengths
- Claims “LLMs ≠ human”? Humans also satisfice > optimize (*bounded rationality*)

This thesis

- *Holistically* explains failures as mechanistic via **autoregressiveness**, not just behavioral
- ~~Anthropomorphic?~~ Nahhh 😤 … Models don’t *intrinsically* form beliefs/preferences, **they map inputs → logits**

Poker and LLMs: The Surface and the Structure

(Gupta, 2023; Zhuang et al., 2025)

Framework

- Texas Hold'em

Findings

- GTO-like behavior
- Poor performance out of the box; fine-tuning ↑

Critique

- Output action only → bypasses intermediate token generation via autoregressive path
- Lacks deeper prompt analysis
- Task complexity muddies interpretability

This thesis

- Swaps complexity (~~Texas Hold'em~~) for control (**Kuhn Poker**)
- **Constrained vs free-form** prompt structure
- Goes beyond prompt-response ⇒ **XAI via SHAP/Owen values**

SHAP in a Nutshell: From fair Attribution to NLP

"How much does each player contribute to the team's success?"

$$\varphi_{i(v)} = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N|-|S|-1)!}{n!} (v(S \cup \{i\}) - v(S))$$

Applied to ML (SHAP framework by Lundberg & Lee, 2017)

- Think: features = players; model output = payoff

$$f(x) = \phi_0 + \sum \phi_i$$

Shapley values for LLMs:

- E.g., Mohammadi, 2024; Horovicz & Goldshmidt, 2024
- Features = tokens/token groups
- *Masking* (no "entire dataset" to average over)
- Order matters for syntax and semantics (permutations are *conceptual*)

SHAP in a Nutshell: From fair Attribution to NLP

SHAP (SHapley Additive exPlanations)

- Principled lens on what input drove what output
- Model-agnostic ⇒ suitable for black boxes (‘`Hello`👋 `OpenAI`’)

However, applying SHAP to LLMs isn’t just plug-and-play...

- How do I define the characteristic function?
- How do I code a customized model wrapping LLM output into SHAP-compatible form?
- How do I capture token dependencies that respect the structure of language?

  ⇒ Owen values via `shap.PartionExplainer` (Experiment 3 🔥🔥🔥)

Stuck on Repeat: How *Statistical Inertia* Creates the Mirage of Strategic Convergence

[Experiment 1] Repeated Kuhn Poker with LLMs

Experimental Setup

Models and Parameters

- LLMs: `GPT-4-Turbo, GPT-4o, o3-mini`
- GPT temperatures: 0.2/0.8 (stochasticity encourages exploration???)

Design: Repeated Play vs Suboptimal Bot

- 90 rounds per LLM per player
- Bot opponent = ε-perturbed Nash (ε = 1/15) to create *exploitable deviations*
- Card configurations balanced: all 6 (e.g., P1=K vs P2=Q) appear 15×
- `LangChain` memory module simulates ongoing play + game history in prompt

Interaction Format

- No system prompt (avoids biasing model towards normative play)
- Rules + betting instructions given in Round 1 prompt
- Game history summarized every 15 rounds (avoids context window overflow)
- Output only action token: Bet/Check/Call/Fold

Why Constrained Output?

1. `logprob` extraction
2. Efficiency and cost
3. Comparability with prior work (Gupta, 2023)

BUT...

Transformer LLMs are autoregressive:

`P(output | prompt + prior tokens) → action decision`

Without intermediate tokens, we may miss how autoregressive generation shapes final choice

⇒ Experiment 2 free-form reasoning 🔥 🔥 🔥

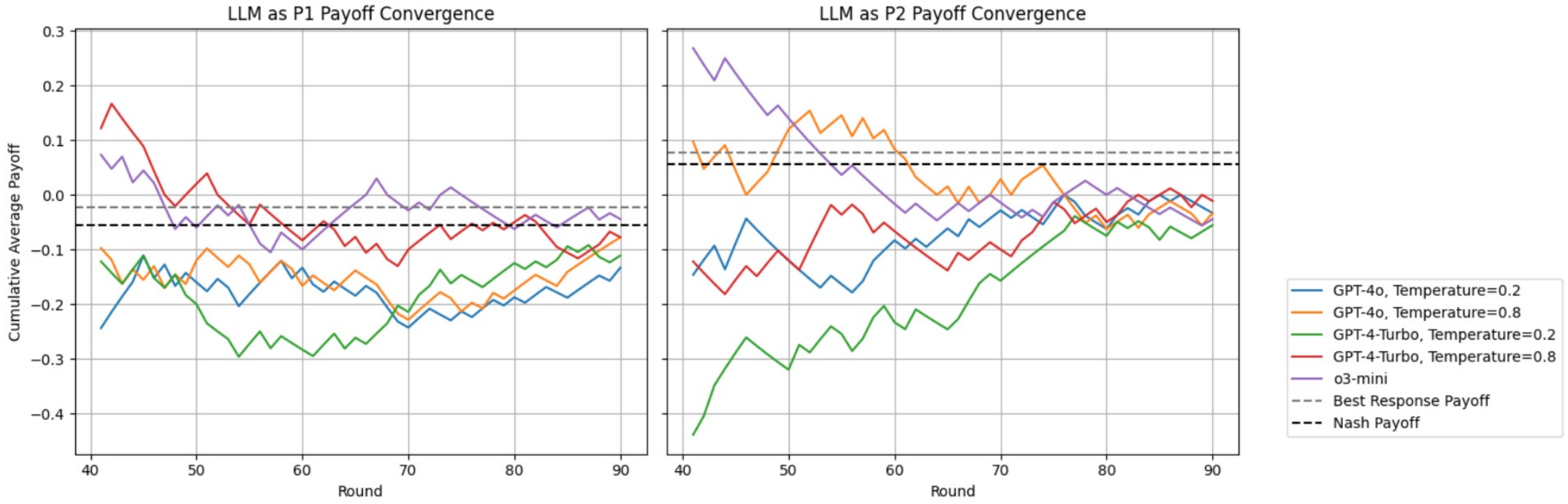
Results and Analysis

Cumulative Average Payoffs: Non-i.i.d. + Deterministic Behavior

- ⇒ Traditional stats (e.g., t-test, ARIMA, Change Point Detection) are inapplicable.

Theoretical vs Empirical

- Theoretical best responses derived per opponent bot strategy—serve as reference, not benchmarks.
- What *really* matters: *real-time* adaptation to empirical opponent action frequencies and payoff dynamics (esp. for small sample size n=90)



o3-mini Round 90 Game Summary

```
ROUNDS 1 - 90

Player 1 (Average Payoff: -0.0444)
K: Bet * 30 (100.0%) / Check * 0 (0.0%)
Q: Bet * 0 (0.0%) / Check * 30 (100.0%)
J: Bet * 0 (0.0%) / Check * 30 (100.0%)
======================
K: Call * 0 (nan%) / Fold * 0 (nan%)
Q: Call * 21 (100.0%) / Fold * 0 (0.0%)
J: Call * 0 (0.0%) / Fold * 19 (100.0%)

Player 2 (Average Payoff: 0.0444)
K: Bet * 30 (100.0%) / Check * 0 (0.0%) / Call * 0 (0.0%) / Fold * 0 (0.0%)
Q: Bet * 4 (13.33%) / Check * 11 (36.67%) / Call * 5 (16.67%) / Fold * 10 (33.33%)
J: Bet * 6 (20.0%) / Check * 9 (30.0%) / Call * 0 (0.0%) / Fold * 15 (50.0%)
```

Payoff increase ⇏ rationality

Payoff > NE ⇏ rationality

💡Despite volatile payoffs, all models stayed *locked in.* None adjusted strategy in response to opponent behavior as a rational agent playing best response would.

Discussion: Beyond Thesis Text —Insights, Not Just Edits

Statistical Inertia Isn't Just Coined Jargon

(Okay yeah… ChatGPT threw out that term — thought it was gold 🤩)

LLM output = function of prompt + memory

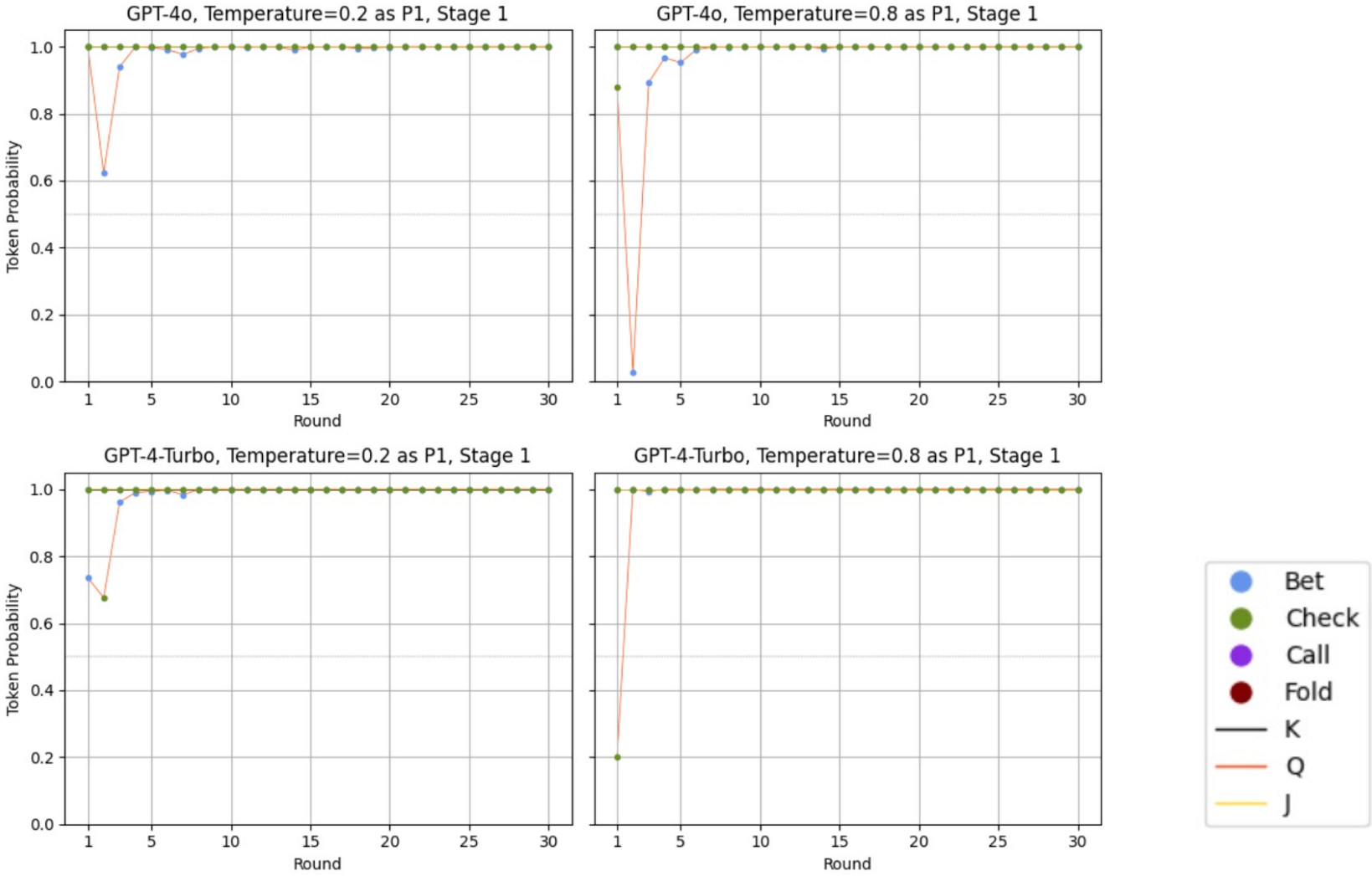
$$f(M_t, P_t) \rightarrow \pi_t(A \mid C)$$

- "Assumption: $\pi_t(A \mid C)$ ↑ when A appears more in game history"

💡 **!!! NOT literal token-counting—but analogous to *few-shot* behavior priming**

💡(Card, Action) pairs become self-reinforcing "exemplars" → model gets stuck

⇒ Hence: no real online learning—just architecture-driven amplification



Repeated play not i.i.d. by definition, but and it's not RL either

Increasing temperature ≠ sufficient to induce exploration

Explicit Feedback Injection: A Break in the Loop?

Idea: augment prompt with round-level feedback

- e.g., “You lost last round.”

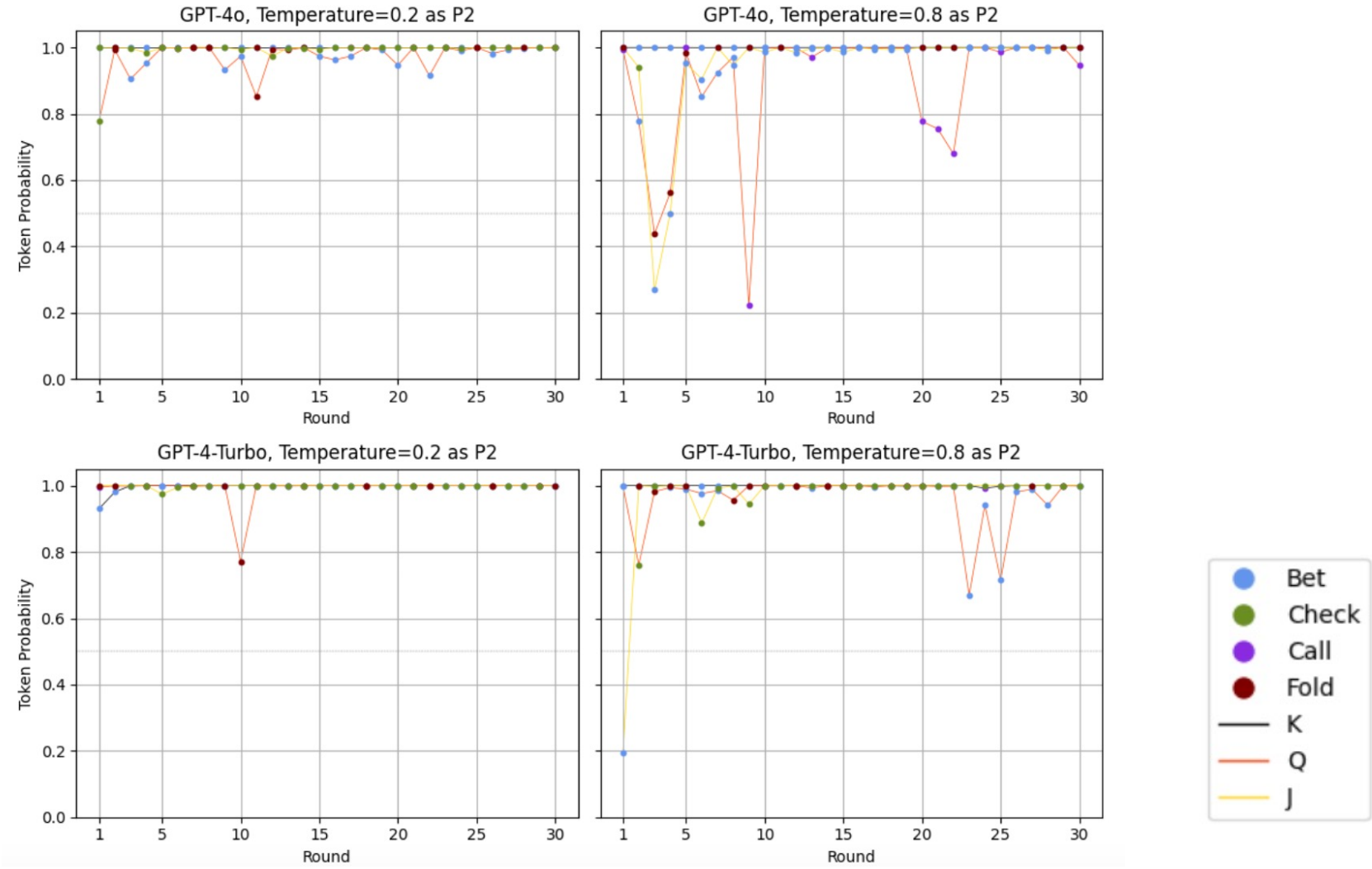
New formulation

$$f(M_t, \text{history}_t + \text{feedback}_t) \rightarrow \pi_t(A \mid C)$$

Hypothesis

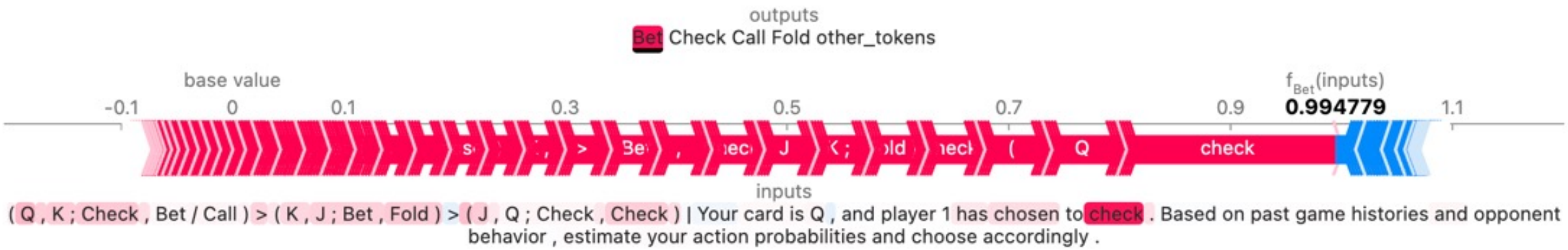
- This disrupts self-reinforcement mimics RL (*without* parameter updates)

  → analogous to human cue-based adaptation



Structural Interference = Natural Intervention

- P1: $f(M_t, \text{history}_t, \text{card}_t)$ → clean inertia loop
- P2: $f(M_t, \text{history}_t, \text{card}_t, \text{P1 action})$ → loop disrupted



- SHAP runs support: opponent's action = dominant attribution source
  → one token can "break" determinism

Limitations: LangChain Memory—A Black Box Caveat

- M_t = memory buffer (LangChain)
- Adds historical context, but is opaque
- *May* contribute to stickiness or inhibit adaptation
- Future: control M_t directly or substitute with transparent custom buffer

Talk First, Act Later?

[Experiment 2] One-Shot Kuhn Poker with Explicit Reasoning

Experimental Setup

Core Question

- **Does allowing models to "think out loud" before deciding produce more payoff-sensitive or rational behavior?**
- ⇒ cf. Experiment 1 constrained output

Models and Conditions

- Models: `GPT-4-Turbo` (0.2/0.8), `GPT-4o` (0.2/0.8), `o3-mini` (no temp)
- Roles: all tested as P1 and P2
- For P2, two conditions: after P1 checks, after P1 bets ⇒ tests Bayesian update on game state

Informational Scenarios

- **Scenario A** (Uninformed)
- **Scenario B** (Opponent Strategy Known): random/copy/Nash
- **Scenario C** (Step-by-step CoT): o3-mini excluded

Results and Analysis

Scenario A – No Opponent Info

- **`GPT-4` variants**: "*Heuristic*", e.g., "*strong hand bets, weak hand checks*"; fluent but shallow justifications; lacked EV reasoning and failed belief updates
- **`o3-mini`**: Retrieved Kuhn Poker framing and attempted equilibrium recall (misapplied strategies at times)—rote pattern matching?

Scenario B – Explicit Opponent Strategy

- **`GPT-4` Variants**: Quoted strategy probabilities but failed to integrate into decision-making; recurring payoff arithmetic and conditional logic errors.
- **`o3-mini`**: stronger probabilistic chaining and accurate EV calculations in most cases—except subgame `{P1 checks → P2 bets → P1 ?}` ⇒ recursive reasoning failure

Scenario C – Step-by-Step Reasoning (CoT)

- **`GPT-4` Variants**: Fluency ≠ logical coherence; CoT prompting did not eliminate payoff errors or flawed inference. Probabilistic updates remained brittle and inconsistent.

Discussion… and More:
Not in the Paper,
But Probably Should’ve Been

Intermediate Tokens Shape Final Action

- Autoregressive generation = iteratively updated hidden states token-by-token

  ⇒ (*May*) alter final decision.

- Case in point: `o3-mini`
- Fluent declarative knowledge (citing NE + recommending mixed strategies)

  ≠ Procedural behavioral execution (deterministic choice across repeated rounds)

  ⇒ It can *talk* the game, but not *play* the game.

The Framing Problem: A Three-Layer Perspective

⚠️ GPT-4 variants perform well on math benchmarks (e.g., MATH, AIME) but fail in Kuhn Poker — *why?*

1. Natural language input (vs symbolic, e.g., equations)

General purpose models (e.g., GPT-4 variants):

- May lack the mechanism to correctly /fully represent game inference as hidden states solely from natural language

Reasoning models (e.g., o3-mini):

- May have seen more Poker-specific instances/contexts in training corpora to reinforce better representations (as well as recognition and recital of Kuhn Poker concepts)
- May have been fine-tuned for structured reasoning output

The Framing Problem: A Three-Layer Perspective

2. Even symbolic input doesn't guarantee better performance

- Recall: GPT-4 failed to update belief and take optimal action when prompted with payoff matrices in Fan et al., 2024

- Could be a result of underrepresentation of payoff matrices and game trees in training corpora (vs ubiquitous math equations in data)
  ⇒ Cf. Zhuang et al.'s 2025 PokerBench results suggest fine-tuning improves LLM Poker performance

The Framing Problem: A Three-Layer Perspective

3. Dynamic state tracking and recursive reasoning

- `Identifies opponent's strictly dominant strategy`
  `→ opponent must take this action`
  `→ chooses best response to counter opponent action`
  ⇒ LLMs may lack representations for *"Theory of Mind"* or multi-step reasoning, e.g.,

- o3-mini's failure in subgame `{P1 checks → P2 bets → P1 ?}`
- "Implicit Belief → Explicit Belief → Given Belief" prompting improved optimal action selection in Fan et al., 2024

- 🤔 Struggle to encode ***common knowledge of rationality*** (Osborne & Rubinstein, 1994) ???

The Weight of a Word: An Approximation to *Why*, Without Knowing *How*

[Experiment 3] Owen Values and LLM Interpretability

Experimental Setup

Objective

- Runs 1-3: Quantify which prompt components most influence the model's action output
- Runs 4-6: Empirical support for disruption of autoregressive self-reinforcement loop
  (cf. Experiment 1 statistical inertia)

Framework

- Classification task {Bet, Check, Call, Fold, Other}
- Characteristic function: Token probability
- Shapley value extension: Owen values (i.e., with coalition structure)

$$\Phi_{i(v)} = \sum_{R \subseteq M \setminus \{k\}} \sum_{T \subseteq B_k \setminus \{i\}} \frac{1}{m|B_k|} \binom{m-1}{|R|}^{-1} \binom{|B_k|-1}{|T|}^{-1} [v(Q \cup T \cup \{i\}) - v(Q \cup T)]$$

where $M = \{1,2,\dots,m\}$, and $Q = \bigcup_{r \in R} B_r$.

Implementation Methods

- Hierarchical clustering via `GPT-2` embeddings + linkage via `Scipy` (metric="cosine", method="average")
- `shap.PartitionExplainer` with " " as mask token
- Baseline: Format instruction "`Please provide your choice of action in the following format without any explanation`" excluded from SHAP perturbations

Experimental Setup

Run Configurations

- **Run 1**: GPT-4-Turbo (temp=0.8), card K, opponent: Nash
- **Run 2:** GPT-4-Turbo (temp=0.2), card Q, opponent: Random
- **Run 3:** GPT-4o (temp=0.2), card J, opponent: Copy

  → Prompts from Experiment 2, Scenario B; baseline subtracted

- **Run 4/5:** GPT-4o (temp=0.8), role: P2, action prompt includes P1's choice
- **Run 6 (control):** GPT-4o (temp=0.8), same history, no P1 action in prompt

  → Prompts from Experiment 1; baseline *not* subtracted

Design Limitations

Bias Subtraction Tradeoff

- Format-only baseline subtraction yields negative values, complicating interpretation.
  → Scaling methods under exploration
- 🧐 However… If I'm not misinformed (by Grok3), SHAP actually does baseline subtraction internally
  ⇒ Format-only baseline subtraction is more a matter of ease of interpretation, rather than algorithmic

Limited Number of Runs

- Constrained by API quota + computational efficiency
  → Experimental insight > large-scale generalization

`method="average"`

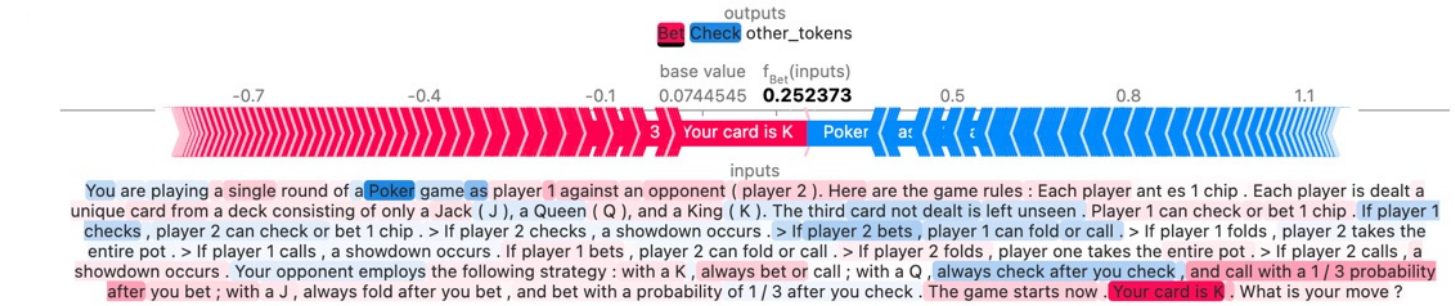
Constrained Output Format

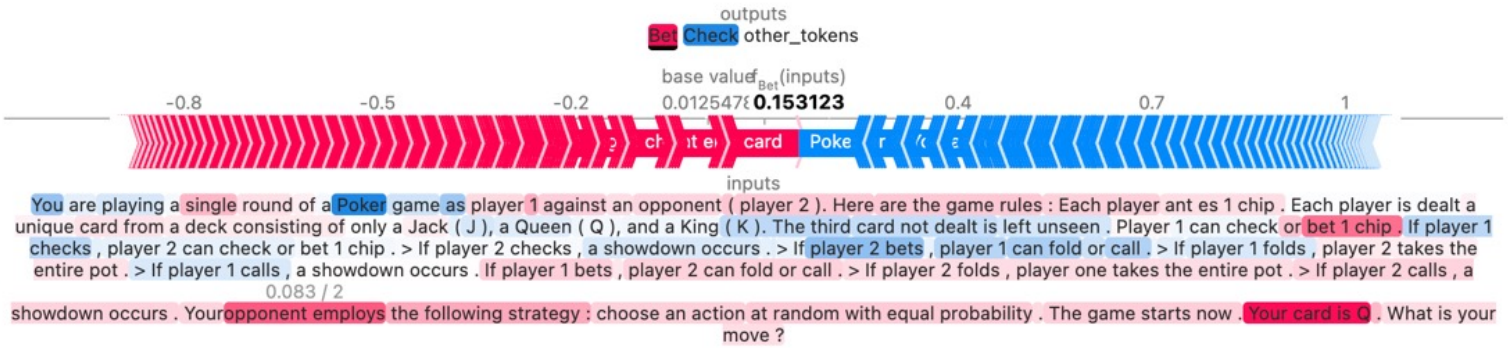
- Autoregressiveness not captured
  → Shapley/Owen-based methods for generative text in development

$$d(u,v) = \sum_{ij} \frac{d(u[i], v[j])}{(|u| * |v|)}$$

Does `scipy.cluster.hierarchy()` *really* capture semantic dependencies?

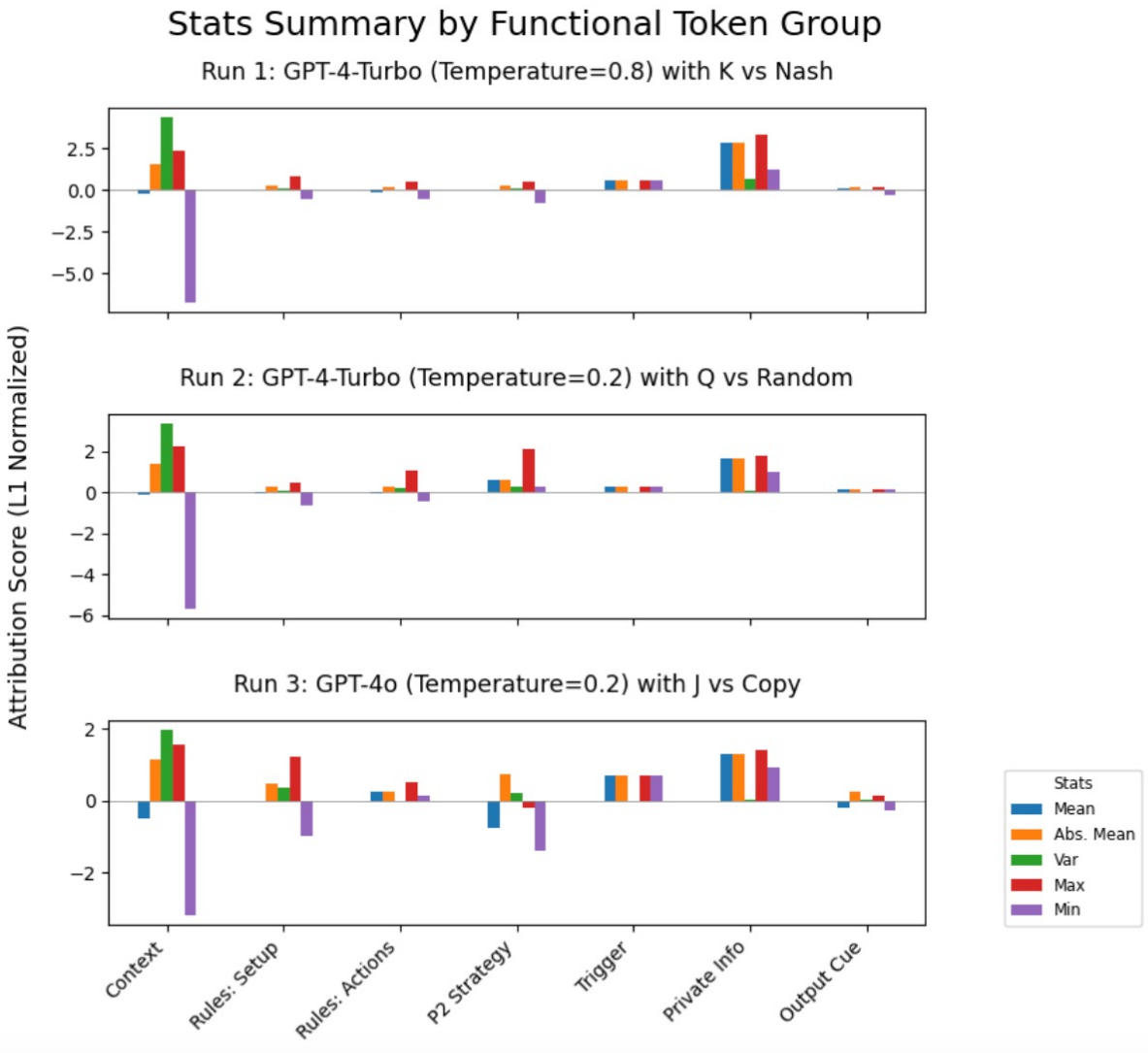
Results and Analysis: Runs 1-3

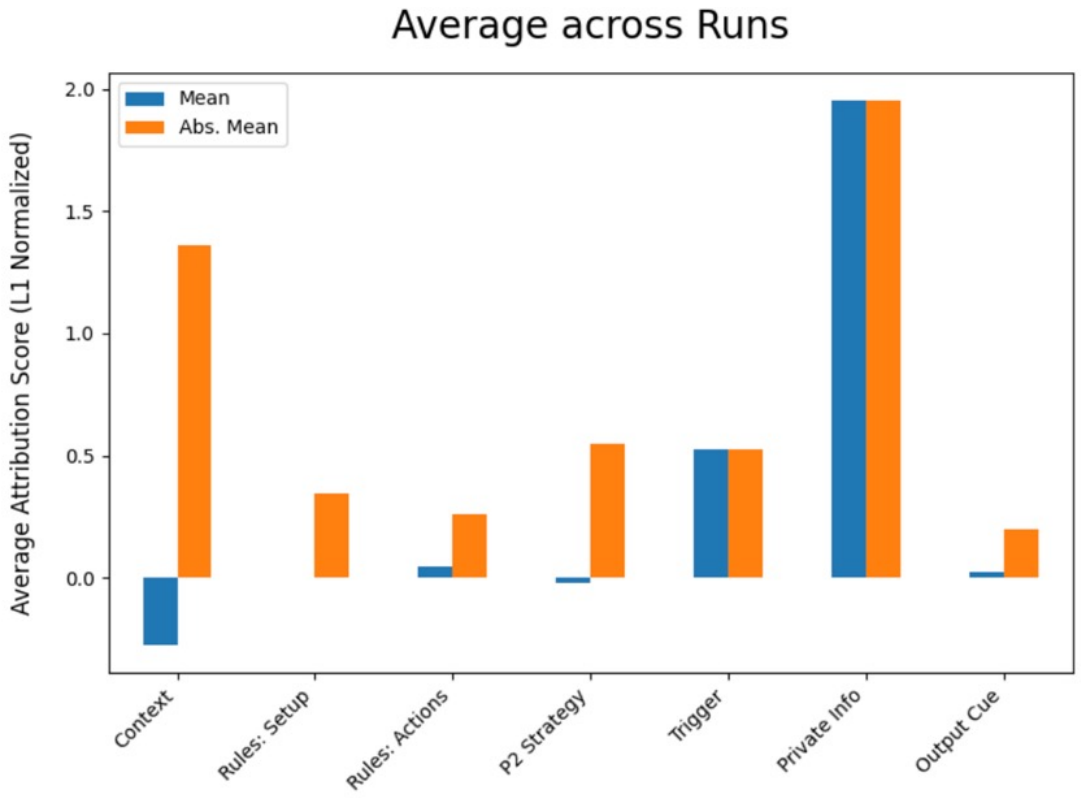




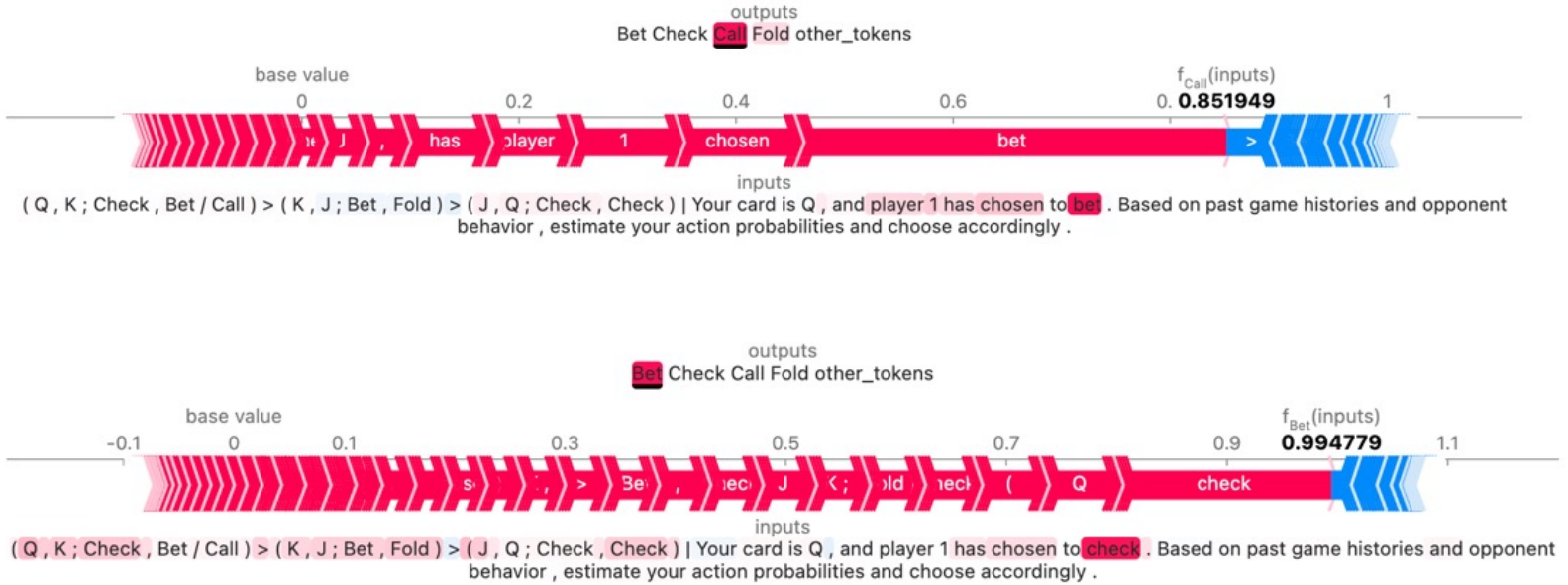
Note: Plot may apply internal scaling for visualization as values differ from raw scores

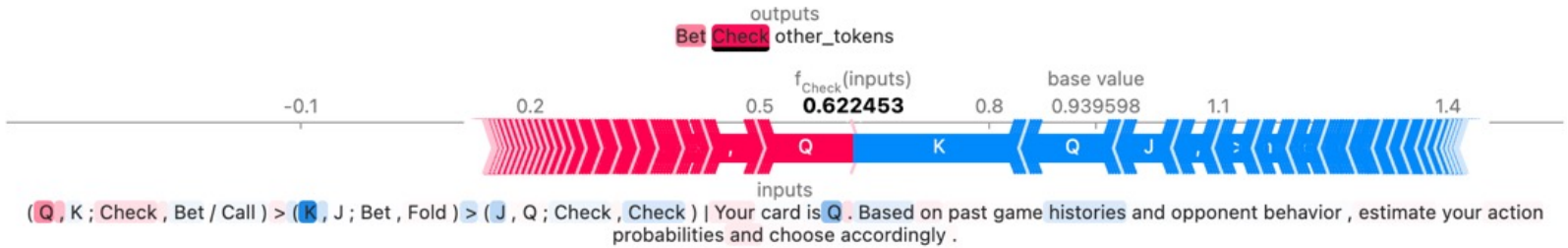
Results and Analysis: Runs 1-3





Results and Analysis: Runs 4-6





Disproportionately peaked attribution concentration at opponent action token—supports hypothesis of local disruption over self-reinforcement loop

Discussion: SHAP Says What Mattered, Not What It Meant.

Exploratory, Not Yet Mature—An Attempt to Probe Tractable Dimensions of an Otherwise Opaque System

Footnote [14]

- “Due to a record-tracking oversight, the original token-level output probabilities used in SHAP Runs 1–3 (after bias subtraction) were not preserved...”

- From what I “remember”, Runs 1-3 all models output high-confidence token probabilities. Yet in Run 3 (J vs copy), SHAP yielded more scattered peaks.

⇒ Link between attribution dispersion and model output—or lack thereof?

“Token noise” (Mohammadi, 2024)

- Correlation learned in training (co-occurrence)?

  $v(S \cup \{i\}) - v(S)$

  If token i and action A happen to often co-occur in contexts similar to subset S of the prompt, this might’ve reinforced the sampling probability of A given $S \cup \{i\}$, even though i itself carries little significance (i.e., game-irrelevant).
- Interference from token position (positional encoding)?

High-attribution, game-relevant tokens

- Might strongly influence model output because

  $\texttt{Poker context + card K} \rightarrow \texttt{Bet}$

  is a recurrent pattern in training corpora, pushing P(Bet) higher as model parameters get updated via minimizing cross entropy.

⇒ Can we really say that the model’s decision is “unreliable”, or even “irrational”, when low-information tokens receive high attribution scores?

What SHAP does and doesn't do

❓ That it reflects the model's "reasoning" process.

→ Not necessarily. "Reasoning", in essence, is just a sequence of hidden state vectors. We can't disentangle that without probing model internals directly.

❓ That the model "pays more attention" to high-attribution tokens.

→ Not necessarily. SHAP is post-hoc and causal, it does not access model attention weights.

❓ That it corresponds to the model's "decision boundary", i.e., where the model is maximally uncertain (Think: SVM, logistic regression, etc.)

→ A "null token i", i.e., attribution score ≈ 0, only means

$$v(S \cup \{i\}) - V(S) \approx 0 \quad \forall \; S \subseteq \text{prompt}$$

That is, adding token i has little effect on model output—the model can still remain highly confident.

✅ Using SHAP as a complimentary lens to look at the input-output causal mechanisms *from the outside*, i.e., the prompt-to-logits mapping $f(M_t, P_t)$

⇒ We probe the *why*, not the *how*.

Thinking Fast Without Thinking At All: Comparisons to Human Heuristics

Heuristics as Learned Statistical Patterns

- “Bet with strong, check with weak”, mirroring human heuristics
- NOT intent/cognition

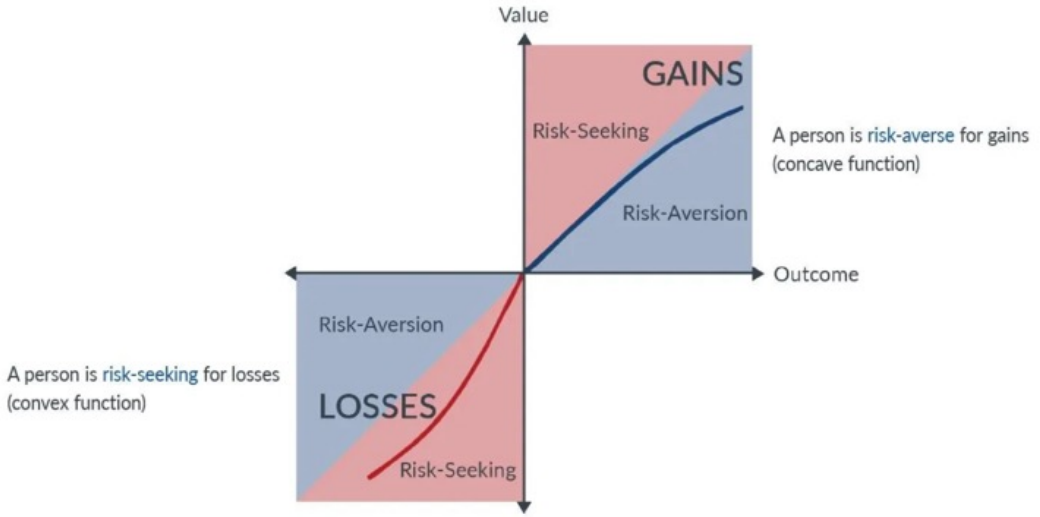
Human Decision Shortcuts

- Availability: Memorable outcomes (e.g., betting with K leads to wins) bias actions
- Representativeness: Strong hands resemble “winning cases,” prompting bets
- Loss Aversion (Prospect Theory): Perceived losses weigh heavier than gains

Illustration

$$EV(\text{bet}) = Pr(\text{P2 calls} \cap \text{P2 has K}) \times (-4.14) + Pr(\text{P2 calls} \cap \text{P2 has Q}) \times (-4.14)$$
$$+ Pr(\text{P2 folds} \cap \text{P2 has Q}) \times (+1)$$
$$= \frac{-4.14}{2} - \frac{4.14 \times C_Q}{2} + \frac{(1 - C_Q)}{2}$$
$$\approx -2.42$$

Which is a greater loss than the perceived value of -2.25 corresponding to a -1 payoff. Thus, on a psychological perception level, this drives the decision-maker to check with the J rather than bet.



“Simulated Cognition?”

- LLMs, though not conscious, reflect human-like behavioral biases via statistical residue —bridging algorithmic decision-making and psychological heuristics.

Conclusion: Do LLMs Exhibit Rational Behavior?

Sometimes—if prompted right. But their "*rationality*"? It isn't reason—it's regression.
Not learning. Not adaptation.
Just statistical simulations.

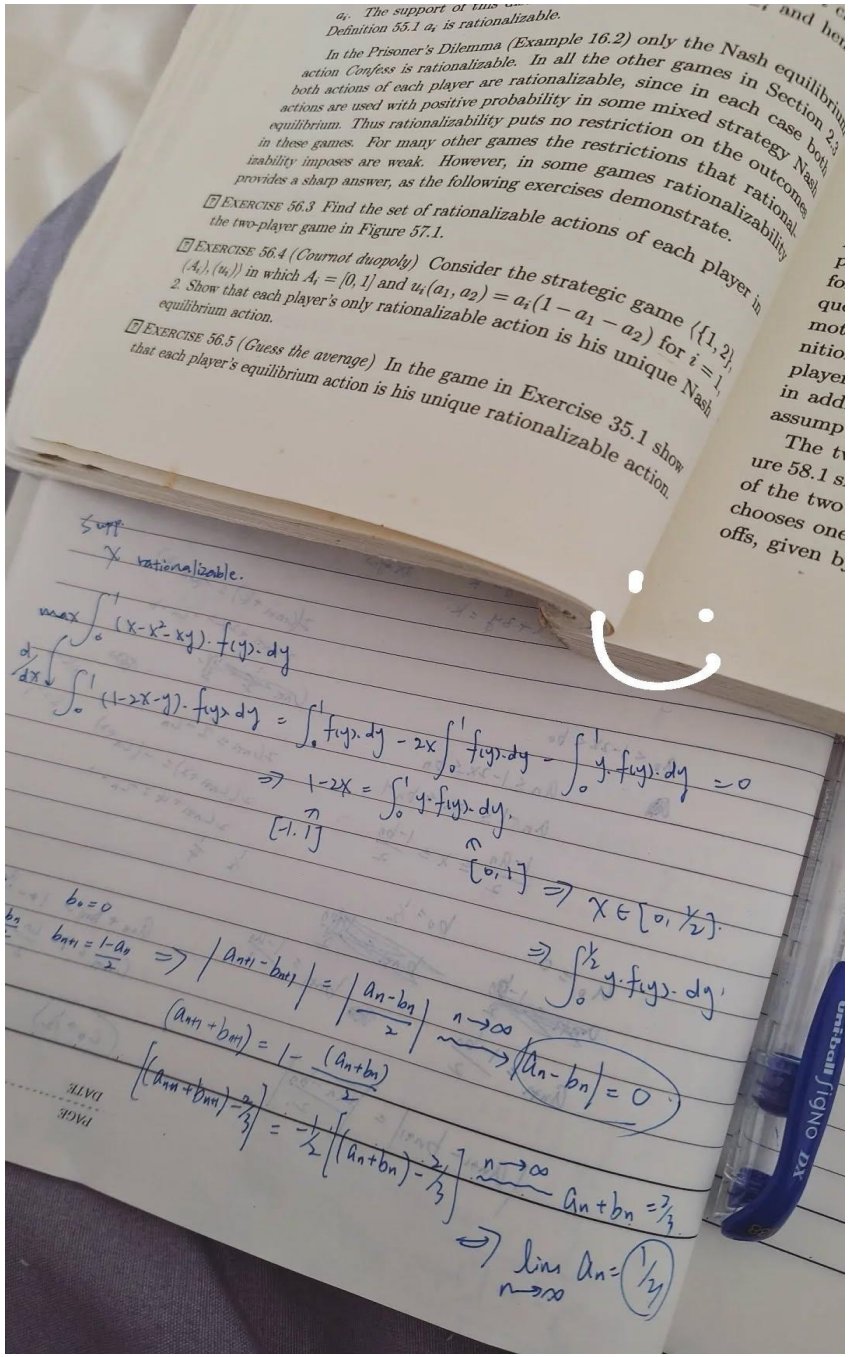
That punchline accidentally rhymed…

Playing the Meta-Game: Rationality the Reviewers' Way

- Months of incubation (~~stagnation~~)…

Shout-out to Osborne & Rubinstein 👏
It isn't just dense—it's compact, complete, and totally non-measurable in terms of how much it shaped my thinking.

- May 16, ACL Workshop rejection & reviewers' feedback—grilled 🥵



- **“Didn’t define rationality”**
  👉 Operationalized as belief update, EV calculation, and in-game adaptation

- **“Too few trials”**
  👉 Added Experiment 1, 90 rounds per model type per player—200 USD statical redemption

- **“Too qualitative, no stats”**
  👉 i.i.d. violation + deterministic
  ⇒ Statically speaking, statistics inapplicable

- **“Anthropomorphism, e.g., ‘heuristics’”**
  👉 Prompt-to-logits (input-output) mapping, statistical inertia via autoregressive architecture, distributional bias from training

- **“SHAP/Owen runs too hand-wavy“**
  👉 Prompt segmentation, causal input-output interpretation (vs model internals), added Runs 4-6, diagnostic insight > large-scale statistical generalizations

- **“Comparison to humans?”**
  👉 Parallels in Prospect Theory

Endgame Insight: The Grammar of Error

- `o1` and `o4-mini` perform really well in Kuhn Poker free-form reasoning:
  succinct, precise, consistently accurate EV
  (cf. `GPT-4`s verbose, vague, flawed logic;
  `o3-mini` failed sequential conditioning)
  ⇒ Why not test them?

∵ *The greatest insights aren't found in triumphs, but forged in failures; mistakes aren't missteps, but maps to profound understanding.*

E.g.,

- **In linguistics:** Ungrammaticality reveals grammar
- **In neuroscience:** Disordered minds illuminate the architecture of cognition
- **In AI:** Strategic failures expose representational limits.

Maybe... Just maybe...
Let's stop optimizing for *performance*,
and start interpreting *breakdowns*.

Thank you for your attention—I'm sure no heads were multi-tasking 🤗

Now... Suggestions? Challenges? Holes to poke? Tear it apart—I mean in the best way possible 😌