PREDICTING DECISIONS OF AI AGENTS FROM LIMITED INTERACTION THROUGH TEXT-TABULAR MODELING

Anonymous authors
Paper under double-blind review

ABSTRACT

AI agents increasingly interact with unfamiliar black-box counterparts whose behavior emerges from language models, prompts, memory, and control logic. We study whether a counterpart's next decision can be predicted from a small set of observed interactions, without access to its implementation. We cast this problem as target-adaptive text-tabular inference, combining behavioral examples from the specific target with the current strategic state, the dialogue, and a representation from a frozen language model Observer. Across bargaining and negotiation environments and a diverse population of independently built agents, adaptation to the target improves prediction in ways that additional population data cannot reproduce. Observer representations provide complementary information beyond explicit state and generic text features, but their value depends on what the structured interaction history already captures. These findings separate learning the behavior of a particular agent from representing the strategic situation, and provide a practical framework for predicting unfamiliar agents from limited interaction.

1 INTRODUCTION

An AI buyer, seller, or procurement assistant may repeatedly face a counterpart it has never seen before. Model identity alone is not enough to predict that counterpart: its behavior can emerge from an underlying LLM combined with private prompts, memory, hand-written rules, tool use, and control code. The practical forecasting target is therefore not average behavior over a model family, but the next decision of one particular black-box agent. We ask how accurately such a counterpart can be predicted after only a small number of observed games.

We call the counterpart being modeled the *target agent*. The predictor observes K complete prior games of that agent, then receives a representation of a new decision point and predicts either whether the target accepts an offer (*response prediction*) or which offer it makes next (*proposal prediction*); Figure 1 illustrates both tasks. We use controlled bargaining and negotiation games to vary incentives and information while retaining ground-truth actions and matched strategic conditions. Both main families use alternating offers, with different payoff and state structures; we do not treat them as evidence of broad protocol-level generalization.

Predicting another agent is a long-standing problem in opponent modeling; we do not claim otherwise (Albrecht & Stone, 2018; Baarslag et al., 2016). We study a specific regime in which source and target agents are independently constructed, the target implementation is unobserved, and no preference model, utility family, acceptance threshold, concession strategy, or opponent type is selected in advance. The task is next-decision prediction under a source-to-target population shift, rather than recovery of a predefined latent opponent parameter.

The final predictor is TabPFN, used as a pretrained in-context tabular model with no task-specific gradient updates. Its conditioning set combines labeled decision rows from source agents with all eligible rows from the target's K observed games. Each row contains structured simulator fields, a generic representation of the current-round messages, an agent-identity indicator, and optionally a decision-oriented hidden state from a small frozen *LLM-as-Observer*. The Observer reads the current decision-time prompt and dialogue only; it does not receive the target's K games, and its answer is not the prediction. This separates representation of the current strategic situation from adaptation to the particular target.

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.

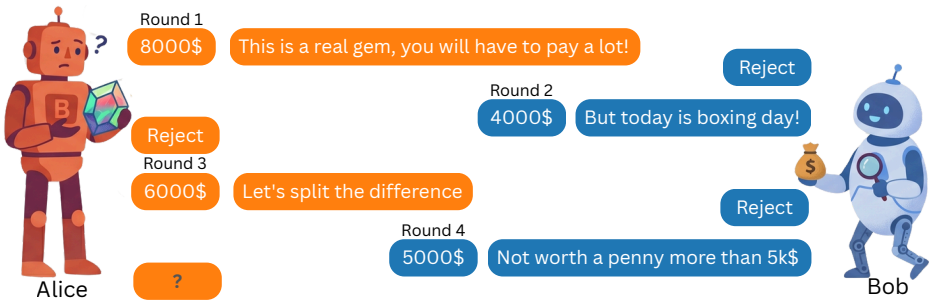


Figure 1: Alice (seller) and Bob (buyer) negotiate via free-text offers. Following Bob's $5,000 round-4 offer, Alice's next move is the prediction target. **(a) Response prediction:** will she accept? **(b) Proposal prediction:** if she rejects, what will she propose?

We evaluate from a source population of 13 frontier LLMs, run under common game-facing prompts, to a held-out population of 91 university-hackathon agents that vary in prompts, control logic, scaffolding, and fallbacks around a restricted LLM API surface. Three findings organize the paper. First, correctly labeled target games produce gains that extra source rows, wrong-target games, and shuffled target labels do not reproduce. Second, Observer hidden states add signal beyond explicit variables, generic text representations, and the Observer's own output. Third, their marginal value is conditional: they help response prediction and bargaining proposals, whereas an explicit trajectory nearly exhausts the signal for negotiation proposals.

Contributions. First, we formulate few-game prediction of black-box language agents as target-adaptive text-tabular inference. Second, we introduce a cross-population evaluation from frontier-LLM agents to independently scaffolded target agents and show that adaptation depends on correctly labeled behavior from the specific target. Third, we characterize what frozen Observer representations contribute beyond explicit strategic variables, generic dialogue encoders, and direct Observer outputs, including settings in which those representations become redundant.

2 Related Work

Language-based strategic agents. LLMs increasingly act in bargaining and negotiation (Shapira et al., 2024b; Xia et al., 2024; Kwon et al., 2024; Bianchi et al., 2024), persuasion (Bao, 2026; Campedelli et al., 2024; Shapira et al., 2024a; 2025), auctions and markets (Chen et al., 2023; Fish et al., 2024; Zheng et al., 2026), and social-agent benchmarks (Zhou et al., 2024; Xie et al., 2026; Karten et al., 2025). This literature primarily evaluates capabilities, outcomes, or population-level behavioral tendencies. GAMEBoT, for example, decomposes games into verifiable reasoning steps to assess the tested LLM's reasoning and actions (Lin et al., 2025). We instead predict the next action of a particular hidden target agent from a few previous games; the target may combine an LLM with prompts, rules, memory, and control logic.

Opponent modeling and limited histories. Opponent modeling is a broad, established problem spanning type-based reasoning, policy reconstruction, classification, plan recognition, and learned behavioral prediction (Albrecht & Stone, 2018; Nashed & Zilberstein, 2022; Gmytrasiewicz & Doshi, 2005; Albrecht & Ramamoorthy, 2013; Albrecht et al., 2016). Automated-negotiation methods infer preferences, reservation values, acceptance rules, or concession strategies from interaction traces (Baarslag et al., 2016; Coehoorn & Jennings, 2004; Chawla et al., 2022); ad-hoc teamwork predicts unfamiliar teammates (Stone et al., 2010; Mirsky et al., 2022; Ribeiro et al., 2023; Wang et al., 2024); and Theory-of-Mind models and human-choice predictors learn behavioral regularities from short histories (Rabinowitz et al., 2018; Nguyen et al., 2022; 2023; Li et al., 2023; Mu et al., 2026; Xiao et al., 2025; Zhang et al., 2025; Cadilhac et al., 2013; Shapira et al., 2025). We do not claim that predicting an unknown agent is new. Our setting combines independently constructed black-box language agents, no preselected opponent parameterization or type family, mixed structured and free-text observations, few complete target games, and evaluation under a source-to-target population shift.

A direct in-context LLM predictor (Brown et al., 2020; Dong et al., 2024) is a natural comparator for this regime.

Text–tabular prediction. Each decision point combines structured fields—offers, round, payoff parameters, and prior actions—with free-form messages. Tabular foundation models perform in-context prediction from labeled rows without task-specific gradient updates (Hollmann et al., 2023; 2025; Qu et al., 2025), matching the need to combine reusable source rows with a few target-specific rows. Prior text–tabular work studies multimodal AutoML, cross-table transfer, benchmarks, and foundation models for tables containing text (Shi et al., 2021; Mráz et al., 2025; Arazi et al., 2026; Kim et al., 2024; Koloski et al., 2025; Arazi et al., 2025). Our focus is not a new tabular algorithm, but target adaptation under agent-population shift.

Frozen LM representations. Intermediate LM states encode syntactic, semantic, and task-relevant variables (Belinkov, 2022; Conneau et al., 2018; Hewitt & Manning, 2019; Tenney et al., 2019), and intermediate or layer-combined representations can transfer better than final outputs (Peters et al., 2018; Hosseini et al., 2023; Skean et al., 2025). Hidden states may also contain signals not reflected in the generated answer (Gekhman et al., 2025; Orgad et al., 2025). This motivates our matched comparison between the Observer's hidden representation and its own output probability. Unlike standard probing, the representation comes from one frozen model observing a public interaction, while the label is the next decision of another black-box agent.

3 DATA

We instantiate our prediction task in GLEE (Shapira et al., 2024b), a benchmark and simulation framework for two-player, sequential, language-based economic games. In GLEE, agents repeatedly make strategic decisions–such as proposing an offer or accepting/rejecting one–while observing the public interaction history and, in the language condition, exchanging free-text messages. The benchmark fixes the game rules while systematically varying payoff parameters, horizons, information regimes, and communication channels. This makes GLEE a natural source for our task: it preserves key ingredients of language-mediated commerce–private values, monetary incentives, multi-turn offers, and strategic dialogue–while providing controlled conditions and ground-truth agent decisions.

We focus on GLEE's two mixed-motive families most aligned with our prediction setting: bargaining and negotiation. In both, two agents alternate offers accompanied by free-text messages, and each decision point can be represented as a text-tabular row containing the recorded configuration, offer history, decision-time dialogue, and the target agent's next move. This yields our two prediction tasks: response prediction, asking whether the target accepts the current offer, and proposal prediction, asking what offer the target makes next. Both families use the same broad alternating-offer structure. We use them as controlled environments with different payoff and state structure, not as evidence of broad protocol-level generalization.

Bargaining. Two agents divide a fixed sum M over multiple rounds in an alternating-offers game (Rubinstein, 1982). At each round, the proposer suggests a split $(p, 1-p)$ and sends a message; the responder accepts, ending the game, or rejects, allowing the interaction to continue with reversed roles. Delay is costly through per-round discount factors $\delta_1, \delta_2 \in (0, 1]$. Configurations vary in the horizon, the discount factors, and whether each agent observes the other's discount factor. Thus, agents must interpret both offers and language when deciding whether to concede, reject, or counter-offer.

Table 1: Agent populations used for cross-population transfer. GLEE agents form the source population; hackathon agents are held-out targets.

Population	Family	# Agents	# Games	# Decisions	Role
GLEE tournament (Shapira et al., 2024b)	Barg.	13	30,888	97,031	source
	Nego.	13	30,940	99,889	source
University hackathon (ours)	Barg.	90	2,936	8,762	target
	Nego.	60	1,985	2,579	target

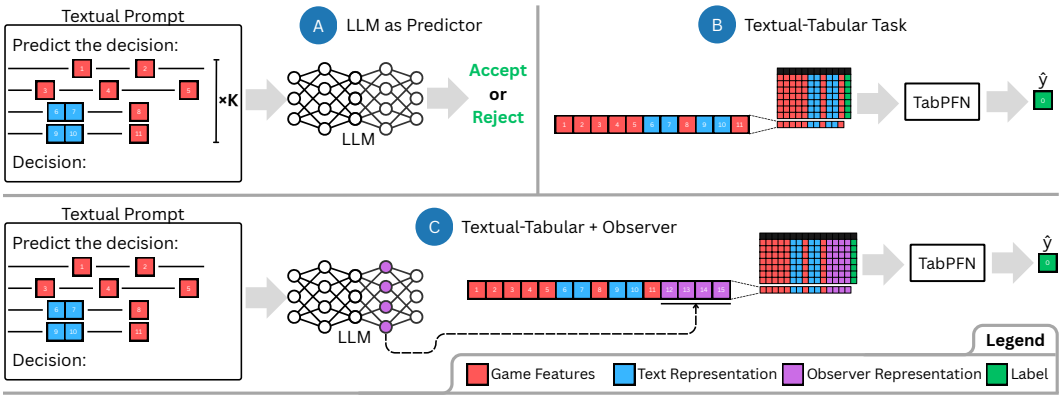


Figure 2: Three prediction interfaces. **(A) LLM-as-Predictor** receives the current interaction and K target games as prompt demonstrations and outputs a decision. **(B) Textual-tabular prediction** represents decisions as labeled rows and uses TabPFN as the final predictor. **(C) Textual-tabular + Observer** appends a hidden-state representation from a frozen Observer that reads only the current decision-time prompt. The Observer never receives the target's K adaptation games; target adaptation and final prediction occur in TabPFN.

Negotiation. A seller with private reserve value V_S and a buyer with private valuation V_B negotiate over the price of a single indivisible good. They alternate price offers, each accompanied by a free-text message. The responder can accept, ending the game; reject and continue when the horizon allows it; or exercise an outside option that guarantees zero surplus. For response prediction, we group outside-option decisions with rejection, since both are decisions not to accept the current offer. Configurations vary in the horizon, valuations, and whether each side observes the other's valuation. Because valuations are private, agents must infer value from offers, signal credibly through language, and decide when agreement remains worthwhile.

We use two complementary agent populations (Table 1): the GLEE frontier-LLM tournament as the source population, where agents vary in the underlying LLM, and a new university-hackathon dataset as the held-out target population, where agents vary in scaffolding around a shared underlying LLM. This split tests transfer between agents constructed along different axes of variation.

Frontier-LLM tournament (source population). The source population is the GLEE round-robin tournament: 13 frontier LLMs from six providers (full model list in Appendix B) play bargaining and negotiation games under identical system prompts, so agents vary only in the underlying model. The tournament covers 960 configurations over horizons, discount factors, valuations, information regimes, and communication regimes, yielding ≈64K games and 197K accept/reject decisions.

University hackathon (held-out target). The target population is a new dataset from a competitive university hackathon held in December 2025, where 34 teams competed for a $2,000 prize. In contrast to the GLEE tournament, agents were restricted to the Gemini 2.5 Flash/Flash-Lite API surface but differed in scaffolding: engineered control logic, prompting pipelines, rule-based fallbacks, or combinations of these. We include logs from all competition stages, treating each submitted team-stage version as a distinct agent, yielding 91 agents, 4,921 games, and 11,341 decisions.

This source–target design tests whether a predictor conditioned on agents that differ mainly in their underlying LLM transfers to agents that differ mainly in scaffolding.

4 Method

The final predictor is TabPFN v2.6, used without task-specific gradient fitting. For each held-out target and each K, TabPFN conditions on labeled source-agent rows together with labeled decision rows extracted from the target's K observed games. A frozen LLM-as-Observer constructs an optional representation of the current public decision state; it does not receive the adaptation games and does not make the final prediction. Figure 2 separates these roles.

4.1 PREDICTION SETTING

At each round, the target agent makes one of two types of decisions. In response prediction, the target receives an offer and must decide whether to accept it. This is a binary classification task. In proposal prediction, the target makes the next offer. This is a regression task over a normalized offer value. Together, these two tasks cover the main observable moves made by agents in bargaining and negotiation games.

For a new target agent, we are given K previously observed complete games and must predict its decisions in held-out games. The target itself is never queried at inference time, and we never observe its prompt, code, or control logic. The Observer and direct LLM baselines receive the reconstructed decision-time prompt. All prediction inputs, including the structured game-state features, are constructed from the deciding agent's player-specific observable view. In incomplete-information games, opponent-private payoff parameters and features derived from them are masked. Splits are made at the game level, so no decisions from one game appear on both the conditioning and test sides.

4.2 FEATURE MODALITIES

Each decision point is represented by three complementary modalities: structured game-state features, a generic dialogue representation, and the Observer hidden-state representation (Figure 4). Together they form a single multimodal tabular row that the predictor of Section 4 consumes.

- **Game-state features.** These features encode the player-observable portion of the simulator configuration, current offer, round index, and previous offers and decisions. The deciding player's own payoff parameter is retained; an opponent's private parameter and any field derived from it are masked whenever the information regime does not reveal them.
- **Dialogue representation.** We concatenate the message fields exchanged in the current round, encode them with MiniLM, and reduce the representation before passing it to TabPFN. This generic representation captures the language accompanying the current decision but is not trained to predict the target's strategy.
- **Observer representation.** The Observer is a small frozen LLM that reads the public decision-time state and available dialogue. It is prompted toward the decision the target is about to make, but its answer is discarded. We instead use an internal hidden state as a decision-oriented representation of the public strategic situation. The Observer is never fine-tuned, never sees target internals, and does not receive the target's K past games. Adaptation occurs only through TabPFN's labeled conditioning rows.

Because the conditioning set mixes rows from many source agents with rows from the current target agent, identical game states may correspond to different behavioral policies. The agent-identity indicator tells the tabular predictor which rows come from the same decision-maker, allowing it to distinguish population-level regularities from target-specific deviations. Without this marker, source and target rows are treated as exchangeable observations even when they are generated by different agents.

4.3 TABULAR PREDICTOR

Each evaluation cell uses default TabPFN v2.6 settings and no task-specific parameter updates. The conditioning set contains up to $3{,}000$ source-agent decisions, balanced across source agents, together with all eligible decisions from the target's K adaptation games. Rows concatenate family-specific recorded game features, MiniLM embeddings reduced from 384 to 5 PCA dimensions, a one-hot indicator over source agents and the current target, and, for Observer variants, a hidden state reduced to 10 PCA dimensions. PCA is fit independently within each evaluation cell on conditioning rows only and applied unchanged to test rows. TabPFN runs in classification mode for responses and regression mode for proposals.

4.4 BASELINES AND CONTROLS

We evaluate the full target-adaptive text-tabular model against direct prompting and reduced tabular baselines, and then isolate the marginal contribution of the Observer feature block.

- **Game+text features baseline.** This baseline uses the same tabular predictor and the same target-adaptation protocol, but removes the Observer representation. It receives only the structured game-state features, the dialogue representation, and the agent-identity indicator. This tests whether the Observer adds information beyond a strong tabular model.
- **LLM-as-Predictor.** This baseline prompts a frontier LLM with the current game, the dialogue, and the target's K observed games, and asks it to predict the target's next decision directly. This tests the natural alternative of using a large LLM as the predictor itself. The approach can read the dialogue and reason over examples in context, but it must commit to a prediction from the prompt alone and does not produce a reusable representation that can be combined with labeled source-population rows.

5 Experimental setup

Evaluation protocol. Our main evaluation is cross-population transfer. We condition on labeled decisions from the frontier-LLM source population and evaluate on held-out hackathon agents, one target at a time. The source population varies the underlying LLM while holding game-facing prompts fixed; the target population uses a restricted LLM API surface while varying prompts, control logic, and rule-based fallbacks.

For each target, we sample $K \in \{0, 2, 4, 8, 16\}$ complete games for adaptation and evaluate on remaining games; all decisions from one game stay on the same side of the split. Results use five seeds and at most 500 test decisions per cell. Response prediction uses AUC, and proposal prediction uses R^2 over normalized offers. We evaluate Gemma-2-2B, Qwen3-1.7B, and Llama-3.2-1B as frozen Observers. For each Observer, Table 2 averages metrics from separately evaluated layers in a prespecified relative-depth band of 0.6–0.9; hidden states are not averaged across layers. The direct baseline uses Gemini 2.5 Flash, the API model available to hackathon participants. Its response run was extended from the initial $65/33$ targets to the same 72 bargaining and 39 negotiation target pool used by the tabular methods; Appendix H additionally recomputes all response rows on the exact target–K–seed cells with defined Gemini AUC.

6 Results

We evaluate cross-population transfer from the 13-agent frontier-LLM tournament to the held-out hackathon agents. The results are organized around two questions. First, does a target-adaptive tabular predictor outperform direct LLM prompting? Second, does the Observer hidden state add information beyond game+text features? All tabular methods use the same TabPFN predictor, with *the Observer* referring to the configuration using game features, text embeddings, and LLM hidden states. Specific models are named by their LLM (e.g., *Gemma-as-Observer*), while *Game+text features* is the baseline without hidden states. The only non-tabular method is *LLM-as-Predictor*, which prompts a frontier API directly. Throughout, K is the number of adaptation games, and performance is measured by AUC (response) and R^2 (proposals).

Response prediction: Observer hidden states are the strongest predictor. Table 2 shows that LLM-as-Observer improves response prediction across both game families and all values of K. We base direct point differences on the exact matched-cell comparison in Appendix H, rather than subtracting summaries with different numbers of defined cells. At $K{=}16$, the best Observer is $+3.7$pp above game+text features and $+7.5$pp above LLM-as-Predictor in bargaining; in negotiation, the corresponding gains are $+5.3$pp and $+7.0$pp. The same pattern appears already at $K{=}0$, where the Observer improves over the tabular baseline without any target-specific examples.

LLM-as-Predictor is weaker. The LLM-as-Predictor baseline uses a large frontier API model and receives the current interaction and the target's K games directly in context. After matching the target, K, seed, and game split, Observer-augmented TabPFN outperforms direct Gemini prompting in both families at every K (Appendix H). Section 7 separates this overall comparison into target-specific adaptation and representation effects.

Proposal prediction: the Observer helps when structured history is not enough. Table 2 shows a conditional effect. In bargaining, all three Observer variants improve over game+text features

Table 2: Cross-population transfer. All response methods use the same $72/39$ bargaining/negotiation target pool, and all proposal methods use the same $78/20$ pool; Appendix H reports the exact direct-LLM-matched cells. **Response** (left): mean AUC, 5 seeds; Observer entries average metrics from layers evaluated separately at relative depth 0.6–0.9. **Proposal** (right): median R^2 on scale-normalised offers, 5 seeds. ±SE. **Bold**: best per family per K; shading: Observer beats strongest baseline.

		Response (AUC)					*Proposal (R^2)*				
	Method	K=0	K=2	K=4	K=8	K=16	K=0	K=2	K=4	K=8	K=16
Barg.	Game+text features	.621 ±.01	.683 ±.01	.712 ±.01	.748 ±.01	.791 ±.01	.366 ±.08	.407 ±.10	.503 ±.08	.502 ±.07	.622 ±.05
	LLM-as-Predictor	.577 ±.01	.686 ±.01	.726 ±.01	.750 ±.01	.770 ±.01	−2.8 ±.80	−1.6 ±.24	−0.5 ±.19	−0.5 ±.19	−0.3 ±.27
	Gemma-as-Observer	**.666** ±.01	**.719** ±.01	**.754** ±.01	**.793** ±.01	**.831** ±.01	.470 ±.07	.513 ±.07	**.598** ±.06	.635 ±.05	**.676** ±.03
	Qwen3-as-Observer	.655 ±.01	.714 ±.01	.749 ±.01	.786 ±.01	.826 ±.01	**.506** ±.06	**.530** ±.07	.590 ±.05	**.641** ±.04	.673 ±.02
	Llama-as-Observer	.642 ±.01	.695 ±.01	.730 ±.01	.770 ±.01	.812 ±.01	.486 ±.07	.519 ±.07	.596 ±.06	.638 ±.04	.674 ±.03
Nego.	Game+text features	.719 ±.01	.768 ±.01	.765 ±.01	.767 ±.01	.803 ±.01	.552 ±.11	.579 ±.12	.567 ±.11	.723 ±.12	.857 ±.08
	LLM-as-Predictor	.486 ±.01	.684 ±.01	.684 ±.01	.713 ±.01	.785 ±.02	−8.0 ±4.8	−3.2 ±2.9	−0.4 ±.81	.164 ±.21	.506 ±.29
	Gemma-as-Observer	.738 ±.01	.782 ±.01	.794 ±.01	.792 ±.01	.826 ±.01	.551 ±.12	.578 ±.13	**.583** ±.12	**.731** ±.12	.845 ±.09
	Qwen3-as-Observer	**.808** ±.01	**.825** ±.01	**.833** ±.01	**.833** ±.01	**.852** ±.01	.540 ±.12	.573 ±.13	.558 ±.12	.711 ±.11	.852 ±.09
	Llama-as-Observer	.743 ±.01	.774 ±.01	.782 ±.01	.779 ±.01	.810 ±.01	**.555** ±.12	**.582** ±.13	.544 ±.11	.725 ±.12	**.862** ±.09

Table 3: Target-adaptation controls for response prediction. Equal-sized extra source data, another target's games, and shuffled target labels do not reproduce the gain from correctly labeled games of the actual target.

Adaptation setting	K	Bargaining AUC	Negotiation AUC
Real target	0	.662	.731
Real target	16	**.830**	**.811**
Real target, no identity	16	.781	.807
Shuffled target labels	16	.589	.621
Wrong target	16	.703	.721
Extra source rows	16	.661	.711

across K, yielding a median R^2 increase of approximately $+0.05$ at K=16. Using Gemma-2-2B as a representative point, this reduces the typical one-offer error on a nominally \$10,000 split from \$552 to \$473, a 14% reduction. In negotiation, the Observer variants provide no reliable marginal gain. Section 7.3 shows that a trajectory-only representation nearly recovers the full negotiation baseline, while leaving a substantial residual gap in bargaining.

Direct LLM proposal baselines. We test both a temperature-0 point estimate and the median of five temperature-0.7 Gemini samples. These comparisons use the same $78/20$ sender target pools, configuration-level splits, and jointly evaluable cells as the corresponding game+text runs. Both improve with K, confirming that the prompt demonstrations are used, but neither matches game+text TabPFN. At K=16, the sampled baseline reaches median R^2 of -0.19 in bargaining and 0.42 in negotiation, compared with 0.63 and 0.86 for game+text TabPFN; MAE yields the same ordering (Appendix I).

7 What Drives Target Adaptation and Observer Gains?

7.1 Target adaptation is agent-specific

We hold the source rows and test set fixed and vary only the adaptation rows. *Real* uses correctly labeled games from the target; *shuffled* permutes labels over the same target states; *wrong target* substitutes matched games from another held-out team; *extra source* adds the same number of source-population rows; and *no identity* removes the identity feature. This control uses Gemma layer 20, within the prespecified layer band, five seeds, and the target agent as the statistical unit.

Real target games increase AUC by 0.168 in bargaining and 0.080 in negotiation, while extra source rows leave performance near the zero-shot value. Wrong-target rows recover substantially less of the gain, and shuffling the target labels lowers performance below K=0. Removing identity reduces

Table 4: Matched Observer readout control at $K=16$ (response AUC; Gemma layer 20).

Added feature	Bargaining	Negotiation
None	.770	.789
Observer output probability	.793	.796
Observer hidden state	**.828**	**.810**

bargaining performance but retains most of the adaptation gain; its effect in negotiation is small. Thus the K-game improvement uses the particular target's state-dependent behavior rather than merely additional examples, state coverage, or a marginal action rate.

7.2 What information does the Observer contribute?

Hidden state versus Observer output. We compare two readouts of the same frozen Observer while holding the downstream game+text+identity stack fixed. At $K=16$, the Observer's acceptance probability improves modestly over no added feature, whereas its hidden state provides a larger gain (Table 4). In bargaining, hidden states outperform output probability at every K, with paired $p \leq 2 \times 10^{-4}$. The result is not explained by simply feeding the Observer's stated answer to TabPFN.

Generic text encoder strength. Replacing MiniLM with `bge-base-en-v1.5` and increasing the text PCA budget changes the baseline by at most 0.015 AUC and does not eliminate the Observer increment. The hidden-state result is therefore not an artifact of an unusually weak generic sentence encoder (Appendix F).

Structured state versus dialogue. Most response-prediction value comes from the Observer's nonlinear representation of the public structured state. At $K=16$, a state-only Observer recovers $+0.046$ bargaining AUC over the engineered baseline, compared with $+0.050$ for the full input, with no significant difference; in negotiation, state-only matches or exceeds the full representation. Dialogue adds measurable information in zero-shot bargaining: the full Observer exceeds state-only by 0.017 AUC ($p = 0.003$). Dialogue-only representations transfer little without the structured state. We therefore interpret the Observer as a decision-oriented representation of the public strategic situation, with additional task-dependent language signal, rather than as a language-only module.

7.3 When does the Observer help?

For proposal prediction, we construct a trajectory-only representation using round, current and previous offers, previous decisions, valuation bounds, and identity. In negotiation it remains within approximately ± 0.025 R^2 of the full game+text baseline at every K, leaving little residual information for the Observer. In bargaining the same representation remains 0.06–0.12 R^2 below the full baseline, and Observer features help close that gap. The marginal value of the Observer is therefore largest when the explicit interaction trajectory leaves predictive uncertainty; it is conditional rather than universal.

7.4 Limited breadth check: persuasion

We apply the unchanged response-prediction pipeline to persuasion, which does not use alternating offers. The Observer gives a small positive increment at every K in a within-GLEE leave-one-agent-out evaluation. Only five agents provide sufficient data, so this result shows that the pipeline is not intrinsically tied to alternating offers, but it is not a third cross-population benchmark or evidence of broad cross-protocol transfer. Provider and layer robustness are reported in Appendix K.

8 Discussion and conclusion

A few correctly labeled games support prediction of a particular unseen target agent. The central evidence is not the monotone improvement with K alone: extra source rows, another target's games,

and shuffled target labels fail to reproduce the gain. This validates the target-adaptive formulation and separates learning about a counterpart from merely increasing the conditioning-set size.

Frozen Observer hidden states contain complementary decision-relevant information beyond explicit variables, generic text encoders, and the Observer's own acceptance probability. The input decomposition narrows this finding: most response-prediction value comes from a nonlinear representation of the public strategic state, while dialogue adds smaller, task-dependent information. Proposal prediction further shows that the representation is useful when explicit trajectories leave residual uncertainty and redundant when they do not.

The evidence is bounded. Bargaining and negotiation are controlled two-player games that share an alternating-offer protocol. We do not establish transfer to real marketplaces, arbitrary protocols, multi-party or multi-issue interactions, or domains without a relevant labeled source population. The persuasion result is preliminary and within-population. Within these limits, the results support separating representation from adaptation: a frozen LLM can enrich the current public state, while a pretrained tabular predictor uses labeled interactions to adapt to a specific unfamiliar agent. The value of the LLM representation is conditional, not universal.

AI USE STATEMENT

We used generative AI tools to assist with writing and editing software code from author-specified designs and experimental protocols, and with language editing of author-written manuscript text. The authors determined the research questions, methodology, experimental design, analyses, and conclusions. The authors verified the reported results and citations and take full responsibility for the final paper and supplementary material.

ETHICS STATEMENT

Predicting an unfamiliar agent's decisions has dual-use implications. Constructive uses include mechanism design and planning under counterpart uncertainty. The same capability could be used to identify and exploit a counterpart's concession patterns. Our target data were collected in a university competition whose participants consented to the research use of the game logs; the released data omit private credentials and retain only interaction artifacts needed for reproducibility.

REPRODUCIBILITY STATEMENT

The appendix specifies the game configurations, source and target populations, feature schemas, conditioning and split protocol, model versions, hyperparameters, prompts, and compute budget. Code and the anonymized target-agent dataset will be included as supplementary material.

REFERENCES

Stefano V Albrecht and Subramanian Ramamoorthy. A game-theoretic model and best-response learning method for ad hoc coordination in multiagent systems. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems*, pp. 1155–1156, 2013.

Stefano V Albrecht and Peter Stone. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence*, 258:66–95, 2018.

Stefano V Albrecht, Jacob W Crandall, and Subramanian Ramamoorthy. Belief and truth in hypothesised behaviours. *Artificial Intelligence*, 235:63–94, 2016.

Alan Arazi, Eilam Shapira, and Roi Reichart. TabSTAR: A tabular foundation model for tabular data with text fields. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=FrXHdcTEzE.

Alan Arazi, Eilam Shapira, Shoham Grunblat, Mor Ventura, Elad Hoffer, Gioia Blayer, David Holzmüller, Lennart Purucker, Gaël Varoquaux, Frank Hutter, and Roi Reichart. MulTaBench: Benchmarking multimodal tabular learning with text and image. *arXiv preprint arXiv:2605.10616*, 2026.

Tim Baarslag, Mark J. C. Hendrikx, Koen V. Hindriks, and Catholijn M. Jonker. Learning about the opponent in automated bilateral negotiation: A comprehensive survey of opponent modeling techniques. *Autonomous Agents and Multi-Agent Systems*, 30(5):849–898, 2016. doi: 10.1007/s10458-015-9309-1.

Michael Bao. ElecTwit: A framework for studying persuasion in multi-agent social systems. *arXiv preprint arXiv:2601.00994*, 2026.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022. doi: 10.1162/coli_a_00422.

Federico Bianchi, Patrick John Chia, Mert Yuksekgonul, Jacopo Tagliabue, Dan Jurafsky, and James Zou. How well can LLMs negotiate? NegotiationArena platform and analysis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 3935–3951. PMLR, 2024.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, volume 33, pp. 1877–1901, 2020.

Anaïs Cadilhac, Nicholas Asher, Farah Benamara, and Alex Lascarides. Grounding strategic conversation: Using negotiation dialogues to predict trades in a win-lose game. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 357–368, 2013.

Gian Maria Campedelli, Nicolò Penzo, Massimo Stefan, Roberto Dessì, Marco Guerini, Bruno Lepri, and Jacopo Staiano. I want to break free! persuasion and anti-social behavior of LLMs in multi-agent settings with social hierarchy. *arXiv preprint arXiv:2410.07109*, 2024.

Kushal Chawla, Gale Lucas, Jonathan May, and Jonathan Gratch. Opponent modeling in negotiation dialogues by related data adaptation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pp. 661–674, Seattle, United States, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.50.

Jiangjie Chen, Siyu Yuan, Rong Ye, Bodhisattwa Prasad Majumder, and Kyle Richardson. Put your money where your mouth is: Evaluating strategic planning and execution of LLM agents in an auction arena. *arXiv preprint arXiv:2310.05746*, 2023.

Robert M. Coehoorn and Nicholas R. Jennings. Learning on opponent's preferences to make effective multi-issue negotiation trade-offs. In *Proceedings of the 6th International Conference on Electronic Commerce*, pp. 59–68, 2004.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2126–2136, Melbourne, Australia, 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1198.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1107–1128, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.64.

Sara Fish, Yannai A. Gonczarowski, and Ran I. Shorrer. Algorithmic collusion by large language models. *arXiv preprint arXiv:2404.00806*, 2024.

Zorik Gekhman, Eyal Ben David, Hadas Orgad, Eran Ofek, Yonatan Belinkov, Idan Szpektor, Jonathan Herzig, and Roi Reichart. Inside-Out: Hidden factual knowledge in LLMs. In *Conference on Language Modeling*, 2025.

Piotr J Gmytrasiewicz and Prashant Doshi. A framework for sequential planning in multi-agent settings. *Journal of Artificial Intelligence Research*, 24:49–79, 2005.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4129–4138, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1419.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular classification problems in a second. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637:319–326, 2025.

MohammadSaleh Hosseini, Munawara Munia, and Latifur Khan. BERT has more to offer: BERT layers combination yields better sentence embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 15419–15431, 2023.

Seth Karten, Wenzhe Li, Zihan Ding, Samuel Kleiner, Yu Bai, and Chi Jin. LLM economist: Large population models and mechanism design in multi-agent generative simulacra. *arXiv preprint arXiv:2507.15815*, 2025.

Myung Jun Kim, Leo Grinsztajn, and Gael Varoquaux. CARTE: Pretraining and transfer for tabular learning. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 23843–23866, 2024.

Boshko Koloski, Andrei Margeloiu, Xiangjian Jiang, Blaž Škrlj, Nikola Simidjievski, and Mateja Jamnik. LLM embeddings for deep learning on tabular data. *arXiv preprint arXiv:2502.11596*, 2025.

Deuksin Kwon, Emily Weiss, Tara Kulshrestha, Kushal Chawla, Gale Lucas, and Jonathan Gratch. Are LLMs effective negotiators? systematic evaluation of the multifaceted capabilities of LLMs in negotiation dialogues. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 5391–5413, 2024.

Huao Li, Yu Chong, Simon Stepputtis, Joseph Campbell, Dana Hughes, Charles Lewis, and Katia Sycara. Theory of mind for multi-agent collaboration via large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 180–192, 2023.

Wenye Lin, Jonathan Roberts, Yunhan Yang, Samuel Albanie, Zongqing Lu, and Kai Han. GAMEBoT: Transparent assessment of LLM reasoning in games. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7656–7682. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.378. URL https://aclanthology.org/2025.acl-long.378/.

Reuth Mirsky, Ignacio Carlucho, Arrasy Rahman, Elliot Fosong, William Macke, Mohan Sridharan, Peter Stone, and Stefano V. Albrecht. A survey of ad hoc teamwork research. In *Multi-Agent Systems. EUMAS 2022*, volume 13442 of *Lecture Notes in Computer Science*, pp. 275–293. Springer, 2022. doi: 10.1007/978-3-031-20614-6_16.

Martin Mráz, Breenda Das, Anshul Gupta, Lennart Purucker, and Frank Hutter. Towards benchmarking foundation models for tabular data with text. In *Foundation Models for Structured Data Workshop at ICML*, 2025.

Chunjiang Mu, Ya Zeng, Qiaosheng Zhang, Kun Shao, Chen Chu, Hao Guo, Danyang Jia, Zhen Wang, and Shuyue Hu. Adaptive theory of mind for LLM-based multi-agent coordination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 29608–29616, 2026. doi: 10.1609/aaai.v40i35.40204.

Samer B Nashed and Shlomo Zilberstein. A survey of opponent modeling in adversarial domains. *Journal of Artificial Intelligence Research*, 73:277–327, 2022.

Dung Nguyen, Phuoc Nguyen, Hung Le, Kien Do, Svetha Venkatesh, and Truyen Tran. Learning theory of mind via dynamic traits attribution. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pp. 954–962, 2022.

Dung Nguyen, Phuoc Nguyen, Hung Le, Kien Do, Svetha Venkatesh, and Truyen Tran. Memory-augmented theory of mind network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11630–11637, 2023. doi: 10.1609/aaai.v37i10.26374.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *The Thirteenth International Conference on Learning Representations*, 2025.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2227–2237, New Orleans, Louisiana, 2018. Association for Computational Linguistics.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 50817–50847. PMLR, 2025.

Neil Rabinowitz, Frank Perbet, Francis Song, Chiyuan Zhang, S. M. Ali Eslami, and Matthew Botvinick. Machine theory of mind. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4218–4227. PMLR, 2018.

João G. Ribeiro, Gonçalo Rodrigues, Alberto Sardinha, and Francisco S. Melo. TEAMSTER: Model-based reinforcement learning for ad hoc teamwork. *Artificial Intelligence*, 324:104013, 2023. doi: 10.1016/j.artint.2023.104013.

Ariel Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50(1):97–109, 1982.

Eilam Shapira, Omer Madmon, Roi Reichart, and Moshe Tennenholtz. Can LLMs replace economic choice prediction labs? The case of language-based persuasion games. *arXiv preprint arXiv:2401.17435*, 2024a.

Eilam Shapira, Omer Madmon, Itamar Reinman, Samuel Joseph Amouyal, Roi Reichart, and Moshe Tennenholtz. GLEE: A unified framework and benchmark for language-based economic environments. *arXiv preprint arXiv:2410.05254*, 2024b.

Eilam Shapira, Omer Madmon, Reut Apel, Moshe Tennenholtz, and Roi Reichart. Human choice prediction in language-based persuasion games: Simulation-based off-policy evaluation. *Transactions of the Association for Computational Linguistics*, 13:980–1006, 2025.

Xingjian Shi, Jonas Mueller, Nick Erickson, Mu Li, and Alexander J. Smola. Benchmarking multimodal AutoML for tabular data with text fields. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021.

Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Nikul Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. Layer by layer: Uncovering hidden representations in language models. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 55854–55875, 2025.

Peter Stone, Gal A. Kaminka, Sarit Kraus, and Jeffrey S. Rosenschein. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24, pp. 1504–1509, 2010. doi: 10.1609/aaai.v24i1.7529.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4593–4601, 2019. doi: 10.18653/v1/P19-1452.

Caroline Wang, Arrasy Rahman, Ishan Durugkar, Elad Liebman, and Peter Stone. N-agent ad hoc teamwork. In *Advances in Neural Information Processing Systems*, 2024.

Tian Xia, Zhiwei He, Tong Ren, Yibo Miao, Zhuosheng Zhang, Yang Yang, and Rui Wang. Measuring bargaining abilities of LLMs: A benchmark and a buyer-enhancement method. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 3579–3602, 2024.

Yang Xiao, Jiashuo Wang, Qiancheng Xu, Changhe Song, Chunpu Xu, Yi Cheng, Wenjie Li, and Pengfei Liu. Towards dynamic theory of mind: Evaluating LLM adaptation to temporal evolution of human states. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24036–24057, 2025.

Sixiong Xie, Zhuofan Shi, Haiyang Shen, Yun Ma, and Xiang Jing. M3-BENCH: Process-aware evaluation of LLM agents' social behaviors in mixed-motive games. *arXiv preprint arXiv:2601.08462*, 2026. URL https://arxiv.org/abs/2601.08462.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Zhining Zhang, Chuanyang Jin, Mung Yao Jia, Shunchi Zhang, and Tianmin Shu. AutoToM: Scaling model-based mental inference via automated agent modeling. *arXiv preprint arXiv:2502.15676*, 2025. URL https://arxiv.org/abs/2502.15676.

Yushuo Zheng, Huiyu Duan, Zicheng Zhang, Yucheng Zhu, Xiongkuo Min, and Guangtao Zhai. Market-Bench: Benchmarking large language models on economic and trade competition. *arXiv preprint arXiv:2604.05523*, 2026.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and Maarten Sap. SOTOPIA: Interactive evaluation for social intelligence in language agents. In *International Conference on Learning Representations*, 2024.

A GAME CONFIGURATIONS

This section lists the exact game configurations played by each population, in support of Section 3. The hackathon target population is restricted to configurations with free-text messages enabled (`messages_allowed`=true); the GLEE source population includes both communication regimes.

GLEE source population. The frontier-LLM tournament sweeps 384 distinct bargaining configurations and 576 distinct negotiation configurations. Bargaining varies the money-to-divide $M \in \{100, 10\,000, 1{,}000{,}000\}$, the round horizon max_rounds $\in \{12, \infty\}$, the information regime (complete vs. incomplete), the communication regime (free-text messages enabled vs. disabled), and the per-player discount factors $\delta_1, \delta_2 \in \{0.8, 0.9, 0.95, 1.0\}$. Negotiation varies the seller's reserve value V_S and the buyer's valuation V_B (each on a relative scale, with both values $\in \{0.8, 1.0, 1.2, 1.5\}$ scaled by a price order $\in \{100, 10\,000, 1{,}000{,}000\}$), the round horizon max_rounds $\in \{1, 10, \infty\}$, the information regime, and the communication regime.

University-hackathon target population. A hackathon configuration uses the same GLEE parameterisation as the source population: in addition to the headline parameters released to participants in advance (horizon, information regime, parameter ranges), each game also fixes per-player payoff parameters that the engine uses to evaluate outcomes—the discount factors δ_1, δ_2 in bargaining and the valuations V_S, V_B in negotiation. The specific values used at evaluation time were not disclosed to teams in advance, so they could not be hard-coded into agent design. At runtime, each agent always observes its own parameter (its own δ in bargaining, its own valuation in negotiation); the opponent's

parameter is observed only under the complete-information regime, and is private otherwise—exactly as in GLEE. All such parameters are recorded in each game's `config.json` and vary across configurations within a stage. We therefore enumerate configurations on the full GLEE parameter set, which yields 10 distinct bargaining and 8 distinct negotiation configurations across the four stages. Table 5 gives the full list.

Table 5: Configurations played at each hackathon stage, enumerated on the full GLEE parameter set (so per-player discount factors δ_1, δ_2 that vary within a stage produce distinct configurations). **Bargaining columns:** M = money to divide; max R = max rounds; info = complete (C) or incomplete (I) information; δ_1, δ_2 = per-player discount factors. **Negotiation columns:** V_S, V_B = seller / buyer relative values; price order = scale; max R, info as above (negotiation configurations do not parameterise per-player discounting). All configurations have `messages_allowed` = true.

Stage	Bargaining					Negotiation			
	M	max R	info	δ_1	δ_2	V_S, V_B	price order	max R	info
1	100	12	C	0.8	0.95	—	—	—	—
				0.8	1.0	—	—	—	—
				0.95	0.95	—	—	—	—
2	10,000	12	I	0.8	0.8	1.0, 1.2	10,000	1	C
						0.8, 1.5	10,000	1	I
3	1,000,000	12	I	0.9	0.9	1.0, 1.5	1,000,000	10	I
Final	100	12	I	1.0	1.0	1.2, 1.0	100	10	I
	10,000	∞	I	0.9	0.8	1.0, 1.2	10,000	∞	I
	10,000	12	C	0.8	1.0	1.2, 1.5	10,000	1	C
	1,000,000	12	I	1.0	0.8	0.8, 1.5	1,000,000	10	I
	1,000,000	∞	C	0.9	0.9	0.8, 1.5	1,000,000	∞	C

B FRONTIER-LLM TOURNAMENT MODEL LIST

The 13 frontier LLMs in the round-robin tournament of Shapira et al. (2024b), grouped by provider.

Provider	Model
Anthropic	Claude 3.5 Sonnet (2024-10-22)
	Claude 3.7 Sonnet (2025-02-19)
Google	Gemini 1.5 Flash
	Gemini 1.5 Pro
	Gemini 2.0 Flash
	Gemini 2.0 Flash-Lite
Meta	Llama-3.3-70B-Instruct
	Llama-3.1-405B-Instruct-FP8
Mistral	Mistral-Large-Latest
OpenAI	GPT-4o
	GPT-4o-Mini
	o3-Mini
xAI	Grok-2 (2024-12-12)

All 13 receive the same game-facing system prompt; the only variable across source agents is the underlying LLM.

C HACKATHON COMPETITION DETAILS

This appendix expands on the held-out target population introduced in Section 3. The hackathon ran in December 2025 over four stages: three preliminary stages with progressively richer game configurations, and a final round in which the top six teams (selected by stage-3 payoff) competed for a $2,000 prize pool. Game configurations released at each stage are listed in Appendix A. To

control API cost across 34 teams playing thousands of games, we restricted participants to the *Gemini 2.5 Flash* and *Gemini 2.5 Flash-Lite* API surface; this choice is what makes the target population orthogonal to the frontier-LLM source on the variation axis (scaffolding instead of underlying LLM).

Released dataset vs. predictive cohort. We distinguish two cuts of the data. The publicly released dataset contains the 23 final-round submissions for which complete code is available for audit. For the predictive evaluation reported in the main text, we additionally retain the decision logs of every (team, stage) agent that appeared at any stage of the competition: teams submitted revised versions between stages, so a single team contributes multiple distinct agents. This yields 91 team-stage agents and 11,341 accept/reject decisions over 4,921 bargaining and negotiation games.

Table 6: Canonical target pools used by each reported evaluation. The direct-LLM runs use the same task-specific pools as the tabular methods. Smaller $K=16$ evaluable counts reflect cells in which the direct baseline has enough held-out data to define the metric, not a separately selected cohort.

Cohort	Bargaining	Negotiation
Raw target agents with family data	90	60
Response comparison target pool (all methods)	72	39
Proposal comparison target pool (all methods)	78	20
Direct LLM with defined $K=16$ response AUC	71	37
Direct LLM with defined $K=16$ proposal R^2	70	19
Persuasion breadth check (within GLEE)	5	

D PROPOSAL PREDICTION: MODEL DETAILS

This appendix documents the model choices specific to the regression task. Aggregate numbers appear in Table 2.

Task and inverse normalisation. For each two-player game in which the target agent plays as proposer, and for each round $r \geq 2$, we predict the offer the proposer will submit–equivalently, the offer its next opponent will face. The regression target is scale-normalised so configurations with different monetary scales share one axis; a closed-form inverse recovers dollar amounts. For bargaining, the target is the proposer's own share of the divided sum; for negotiation, it is price divided by a configuration-specific scale constant. Round 1 is excluded because it has no prior-round state to condition on.

Task-oriented prompt suffix. The reconstructed decision-time prompt used by the response-prediction model ends with `{"decision": "`, which primes the Observer for an `accept`/`reject` token. That suffix is uncorrelated with the proposal regression target. We therefore swap the suffix for a task-matched one that orients the Observer toward the proposer's own next offer:

- Bargaining proposal: `Offer: {proposer_name}_gain: $`—primes for the proposer's own dollar gain, which is our target variable for that family.
- Negotiation proposal: `Offer: $`—primes for the proposer's price amount.

This is the only change at Observer-extraction time. The response-prediction suffix is unchanged from the main model.

Feature stacks. The proposal-prediction comparison in Table 2 contrasts the same two stacks used for response prediction:

- *Game+text features baseline:* the proposer-side game-feature schema (round index, horizon, discount factors, per-side valuations, prior-round offers and decisions), the dialogue representation, and the agent-identity indicator.
- *Observer-augmented model:* the same stack plus the Observer hidden-state representation. This is the row reported under each Observer model in Table 2.

The primary comparison is the Observer's marginal contribution on top of the game+text features baseline.

Cohort filters. We include hackathon agents with at least 30 round-$\geq$2 proposer decisions and target standard deviation ≥ 0.02; bargaining passes 78 agents, negotiation passes 20.

Evaluation. Same protocol as response prediction: cross-population transfer from the frontier-LLM tournament, conditioning sets capped at 3,000 source rows (balanced across source agents) plus all target adaptation rows, test rows capped at 500 per cell, game-level splits, $K \in \{0, 2, 4, 8, 16\}$, 5 seeds, and TabPFN v2.6 in regressor mode.

Why median. Per-agent R^2 in this regression task is heavy-tailed: for some agents, predictions extrapolate to extreme values on a small subset of configurations, driving per-agent R^2 to large negative values. Means over agents are dominated by these tail cases, while medians are stable. We therefore report medians in Table 2 and in the primary discussion of proposal-prediction results.

E TARGET-SPECIFIC ADAPTATION CONTROLS

For each target and seed, we keep the source rows, held-out test games, and number of adaptation rows fixed. *Real* uses the target's own games and labels. *Shuffled* preserves those states but permutes their labels within the adaptation set. *Wrong target* draws a matched number of games from another held-out team and marks them as target rows. *Extra source* draws the same number of additional source-population rows. *Real, no identity* removes the one-hot identity block from both conditioning and test rows. The primary control fixes the Observer at Gemma layer 20, selected from the prespecified band rather than optimized on these results; its bargaining AUC is within 0.002 of the band summary.

At K=16, Real exceeds Extra source by 0.169 bargaining AUC ($p < 10^{-11}$) and 0.100 negotiation AUC ($p < 10^{-4}$). Real exceeds Wrong target by approximately 0.126 and 0.090, respectively ($p \leq 10^{-4}$). Shuffling labels reduces performance below K=0, ruling out an explanation based only on state coverage or marginal acceptance rate. Significance tests are paired over target agents.

F OBSERVER REPRESENTATION CONTROLS

Hidden state versus output probability. The matched control in Table 4 adds either the Observer's renormalized next-token acceptance probability or its 10-dimensional PCA-reduced hidden state to the same game+text+identity rows. In bargaining, the hidden-state advantage over output probability ranges from 0.025 to 0.035 AUC across K, with paired $p \leq 2 \times 10^{-4}$ in every cell. This comparison concerns two readouts of the same frozen Observer and keeps TabPFN fixed.

Generic text representation. We replace `all-MiniLM-L6-v2` with `bge-base-en-v1.5` and separately increase the text PCA budget. BGE changes the game+text baseline by at most about 0.015 AUC, and neither intervention removes the incremental hidden-state gain. The generic encoder processes the same current-round message string described in Appendix N.

G OBSERVER INPUT DECOMPOSITION

We reconstruct each Observer prompt in three forms: *full*, containing the target-observable structured state and dialogue; *state only*, with dialogue removed; and *dialogue only*, with structured and numerical state masked. Reconstructing the original full prompt reproduces the stored representation exactly on sampled decisions (cosine similarity 1.000), validating the reconstruction path. All variants retain the same restriction as the main Observer prompt.

At K=16, state-only recovers nearly all of the full bargaining increment over the engineered baseline (+0.046 versus +0.050 AUC; no significant difference), and it matches or exceeds full in negotiation. At K=0, full exceeds state-only by 0.017 in bargaining ($p = 0.003$), while dialogue-only transfers

little. The decomposition supports a nonlinear-state interpretation with a smaller, task-dependent dialogue contribution.

H MATCHED DIRECT-LLM COMPARISONS

The response baseline was initially run on 65 bargaining and 33 negotiation targets. We subsequently evaluated the missing 7 and 6 targets with the same Gemini 2.5 Flash prompt, decoding configuration, game-level splits, and seeds, producing the unified $72/39$ target pool used in Table 2. The old $65/33$ counts in earlier drafts therefore do not describe the reported LLM row.

Because the API evaluation caps each target–seed test set at 30 decisions, AUC is undefined when that subsample contains only one response class. Table 7 handles this conservatively: for every K, it restricts Game+text and all three Observer variants to the exact target–seed cells for which Gemini AUC is defined. The game split is identical within each matched cell. Thus, the direct differences reported in Section 6 do not subtract estimates from different target cohorts or different random splits.

Table 7: Exact matched-cell response comparison (mean AUC). Every tabular entry is recomputed on the target–K–seed cells with a defined Gemini AUC. The unified target pools contain $72/39$ bargaining/negotiation agents; at $K{=}16$, the defined subsets contain $71/37$. Observer rows average metrics from separately evaluated layers in the prespecified relative-depth band.

Family	Method	$K{=}0$	$K{=}2$	$K{=}4$	$K{=}8$	$K{=}16$
Bargaining	Game+text features	.637	.699	.727	.759	.808
	LLM-as-Predictor	.577	.686	.726	.750	.770
	Gemma-as-Observer	**.682**	**.734**	**.767**	**.806**	**.845**
	Qwen3-as-Observer	.670	.728	.763	.797	.840
	Llama-as-Observer	.658	.711	.745	.780	.827
Negotiation	Game+text features	.708	.761	.760	.763	.802
	LLM-as-Predictor	.486	.684	.684	.713	.785
	Gemma-as-Observer	.734	.777	.789	.789	.820
	Qwen3-as-Observer	**.813**	**.830**	**.836**	**.837**	**.855**
	Llama-as-Observer	.739	.770	.779	.781	.806

The proposal comparison is matched independently to its sender-side task: both direct Gemini variants and the tabular methods use the same 78 bargaining and 20 negotiation target pools and the same configuration-level splits. At $K{=}16$, the jointly evaluable subsets contain 70 and 19 targets, respectively. On these matched evaluations, the five-sample Gemini median obtains $R^2 = -0.19/0.42$, versus $0.63/0.86$ for game+text TabPFN (bargaining/negotiation); the ordering also holds under MAE.

I PROPOSAL-PREDICTION DIAGNOSTICS

Trajectory-only control. The reduced stack contains round index, current and previous offers, prior decisions, player-observable valuation bounds, and agent identity. It stays within approximately $\pm 0.025\, R^2$ of the full negotiation game+text baseline at every K. In bargaining it remains 0.06–$0.12\, R^2$ below that baseline. The diagnostic follows the same player-specific information masking as the main game-state features.

Regression-oriented direct LLM baselines. The temperature-0 baseline returns one Gemini offer. The sampled baseline returns the median of five temperature-0.7 offers from the same prompt and adaptation games. Both use the same sender targets, game splits, and jointly evaluable cells as the corresponding tabular comparison (Appendix H), and both improve with K. At $K{=}16$, the sampled baseline obtains median R^2 of -0.19 in bargaining and 0.42 in negotiation, versus 0.63 and 0.86 for game+text TabPFN. MAE gives the same ordering. Negative R^2 means the predictor is worse than using each target's held-out mean offer under the reported evaluation.

J LIMITED PERSUASION BREADTH CHECK

Persuasion differs from the two main families because it does not use alternating offers. We run the unchanged response-prediction pipeline in a within-GLEE leave-one-agent-out protocol. Only five agents produce enough persuasion decisions, and no independently scaffolded persuasion target population is available. The Observer increment is small but positive at every K; we report it only as a check that the representation pipeline is not mechanically tied to offer exchange.

K FEATURE-BLOCK, LAYER, AND PROVIDER ROBUSTNESS

Table 8: Feature ablation at $K=16$ (Gemma-2-2B). Left: leave-one-out from the full stack; right: reduced stacks. **G**=game, **T**=text, **O**=Observer, **I**=identity. Metrics are averaged over separately evaluated mid-to-late Observer layers.

		Full	−O	−T	−G	−I	G+I	T+I	O+I	I
Response (AUC)	Barg.	.831	.791	.833	.778	.779	.796	.583	.783	.500
	Nego.	.826	.803	.815	.615	.817	.779	.621	.604	.500
Proposal (R^2)	Barg.	.689	.616	.695	.618	.607	.621	.013	.625	−.038
	Nego.	.841	.844	.843	.450	.722	.843	.063	.438	−.057

Structured features are the backbone, especially in negotiation. Observer features add to the full response stack and to bargaining proposals; generic current-round text embeddings are often redundant once the Observer is present.

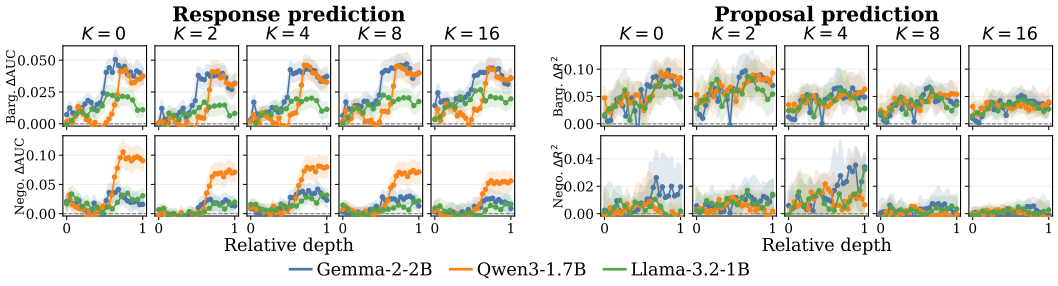


Figure 3: Observer gain over the game+text baseline by relative depth. Each point evaluates one layer; metrics, not activations, are summarized across the prespecified 0.6–0.9 depth band. Left: response ΔAUC; right: proposal ΔR^2.

Gains remain stable across three Observer families and across the predefined mid-to-late layer range. The following appendix reports the matched provider comparison.

L PROVIDER REPLICATION OF THE LOGITS-VS-HIDDEN-STATES GAP

This appendix extends the robustness analysis of Appendix K. We test whether the gap between an Observer's direct logits and its hidden-state representation depends on the choice of provider. The comparison uses the cross-population protocol of Section 6 ($K=16$, TabPFN classifier, 5 seeds) and the main response cohort: 72 bargaining and 39 negotiation targets.

Logits vs. hidden states. Table 9 isolates the effect for the Gemma-2-2B Observer. Logits alone, whether passed through a TabPFN classifier or read directly as $p(\text{accept})$, are far weaker than the Observer hidden-state representation. The game+text features baseline is already strong; adding the logit scalar yields only a marginal improvement, while adding hidden states yields a clearly larger one. Adding both logits and hidden states is a wash relative to hidden states alone. The pattern matches the Section 6 takeaway: the predictive signal lives in the Observer's representation, not in its direct readout.

Table 9: Matched comparison of logits vs. hidden states under cross-population transfer at $K=16$ with TabPFN, mean AUC over 5 seeds (SE over agents $\times$ seeds in parentheses; 72 bargaining, 39 negotiation targets). Logits $= p(\text{accept})$ obtained by renormalising next-token probabilities over the accept/reject verbalisers. Band summaries average metrics from separately evaluated Gemma-2-2B layers at relative depth 0.6–0.9; activations are never averaged across layers.

Input to the TabPFN classifier	Bargaining	Negotiation
Logits alone	.585 (.010)	.464 (.015)
Observer hidden states alone	.701 (.008)	.536 (.013)
Game-state features alone	.756 (.010)	.827 (.014)
Game-state features + logits	.767 (.009)	.831 (.014)
Game-state features + Observer hidden states	**.793** (.008)	**.838** (.013)
Game-state features + logits + Observer hidden states	.791 (.008)	.835 (.013)

Replication across three providers. If the logit-vs-hidden-state gap were an artifact of the hackathon agents' underlying LLM matching the Observer's training pipeline, Observers from unrelated providers should fail to reproduce it. They do not. We replicate with Qwen3-1.7B (Yang et al., 2025) (Alibaba) and Llama-3.2-1B (Grattafiori et al., 2024) (Meta), neither of which shares a training pipeline or parent company with the hackathon agents' underlying LLM, and recover the same pattern (Table 10). Game-state features plus Observer hidden states sit in a tight band across providers (bargaining $\in [0.76, 0.79]$, negotiation $\in [0.84, 0.86]$). Direct $p(\text{accept})$ AUC is more variable across providers, especially in negotiation: Gemma and Llama remain near chance (≤ 0.45) while Qwen3 reaches 0.717, an outlier we report transparently rather than aggregate over. The hidden-state-augmented predictions are consistent across providers; the direct readout is not.

Table 10: Logits vs. hidden states across Observer providers at $K=16$. Logits columns report direct $p(\text{accept})$ AUC. Hidden-state columns use game features plus one layer's representation under the same TabPFN classifier; provider summaries average metrics from separate runs over the prespecified relative-depth band, not the activations themselves.

			Logits alone		*Game + Hidden states*	
Observer	Provider	Params	Barg	Nego	Barg	Nego
Gemma-2-2B	Google	2.6B	.548	.403	.793	.838
Qwen3-1.7B	Alibaba	1.7B	.607	.717	.785	.863
Llama-3.2-1B	Meta	1.2B	.577	.446	.763	.837

M Additional experimental details

Hardware. All GPU experiments ran on an internal cluster with up to 2$\times$ NVIDIA RTX A6000 (48 GB each); individual jobs used a single GPU at a time. CPU-only steps (game-feature extraction and dialogue encoding with `all-MiniLM-L6-v2`) ran on the same hosts. The direct LLM-as-Predictor baseline (Section 6 and Appendix O) is API-only and consumes no local GPU time.

Observer feature extraction. Decision-time hidden states for Gemma-2-2B, Qwen3-1.7B, and Llama-3.2-1B were extracted with TransformerLens `run_with_cache` over reconstructed prompts for $\approx$ 67K games ($\approx$ 200K decisions). Upper-stack layers are cached together, but every layer is evaluated as a separate feature representation; reported band results average downstream metrics across those runs. Three suffix variants orient the Observer toward response, bargaining-proposal, and negotiation-proposal prediction (Appendix D). The same pass caches next-token logits. Cumulative Observer extraction: $\approx$ 80 A6000 GPU-hours.

Tabular evaluation. TabPFN v2.6 was run with default settings and no task-specific parameter updates. Each evaluation cell uses up to 3,000 source-balanced conditioning rows together with the target's K-game decisions and at most 500 test rows, taking $\approx$ 10–30 s on one A6000. The budget covers the cross-population response and proposal tables, separate layer evaluations, and the matched readout controls. Cumulative tabular evaluation: $\approx$ 60 A6000 GPU-hours.

LLM-as-Predictor (API). The LLM-as-Predictor numbers in Tables 2–2 use Gemini 2.5 Flash with `thinking_budget=0`. The response run covers the full 72 bargaining and 39 negotiation target pools after extending the original $65/33$ run; the proposal run covers the same $78/20$ sender pools as the tabular proposal methods. Each run uses $K \in \{0, 2, 4, 8, 16\}$, five seeds, and up to 30 test decisions per cell. Appendix H reports exact target–seed matched comparisons and explains the few cells whose metric is undefined. Because no thinking tokens are generated, the per-call cost is much lower than the pilot. The thinking-on pilot in Appendix O used $4{,}010$ calls with `thinking_budget=2000` at a cost of $\approx$\$31; we did not scale this configuration to the full table, which would have cost an estimated $\approx$\$235.

Total. The experiments reported in the paper consumed ≈ 140 A6000 GPU-hours cumulatively (extraction + tabular evaluation), plus the Gemini API budget above. Including preliminary and discarded runs—alternative Observer layers and pooling strategies, predictor sweeps that did not converge under the cross-population protocol, abandoned feature stacks, and exploratory hackathon-only protocols not reported in the final paper—the broader project consumed roughly ≈ 400 GPU-hours.

Model hyperparameters. TabPFN v2.6 with default settings (no tuning). Observer metrics are averaged over separately evaluated layers in each model's prespecified relative-depth band (0.6–0.9); layer counts are Gemma-2-2B (26), Qwen3-1.7B (28), and Llama-3.2-1B (16). Seeds $\{0, 1, 2, 3, 4\}$. Full feature specifications appear in Appendix N.

Observer prompt suffixes. The suffix appended after the dialogue history orients the Observer toward the decision to predict: `{"decision": "` for response prediction (the original GLEE accept/reject format), `Offer: $` for proposal prediction in negotiation, and `Offer: {proposer_name}_gain: $` for proposal prediction in bargaining (proposer's own dollar gain).

Game-level split. For each held-out target agent, at each K, we randomly select K of the agent's games as the K-shot pool (all round-by-round decisions within those games enter the adaptation rows); the remaining games form the test set. Splitting at the game level ensures that rounds from the same game never appear on both sides of the split, and makes explicit that K counts whole observed games–each with multiple decisions–rather than individual decisions.

Error bars. Throughout the paper, $\pm$ values report standard errors of per-(agent, seed) AUC or R^2 values for each cell ($\sigma/\sqrt{N}$ with $N = N_{\text{targets}} \times 5$). Figure 3 reports paired 95% confidence intervals as $1.96 \cdot \text{SEM}$ on per-(agent, seed) deltas of full $-$ baseline.

N GAME+TEXT FEATURES BASELINE: FEATURE SPECIFICATIONS

This appendix specifies the three feature blocks of the game+text features baseline (Section 4): game-state features, the dialogue representation, and the agent-identity indicator. The Observer hidden-state representation, appended in the Observer-augmented model, is documented in Appendix M.

Game-state features – bargaining (24 columns). At each responder decision at round r we extract a fixed-schema vector from the deciding player's observable game view, with $N{=}5$ prior rounds of history. Missing or opponent-private entries are encoded as `NaN`; TabPFN handles `NaN` natively. The columns are:

- *Configuration (8):* `round`, `max_rounds`, `round_frac` ($=r/$`max_rounds`), `money` (the amount to divide M), `delta_1` and `delta_2` (the players' per-round discount factors, populated only when observable to the deciding player), `messages` (binary: free-text exchange allowed in this game), `complete_info` (binary: parameters shared with both players or private).
- *Current offer (3):* `offer_frac`, `responder_gain`, and `proposer_gain`. The fraction is the responder's gain divided by the sum of both gains; the latter two fields give the split in absolute units.

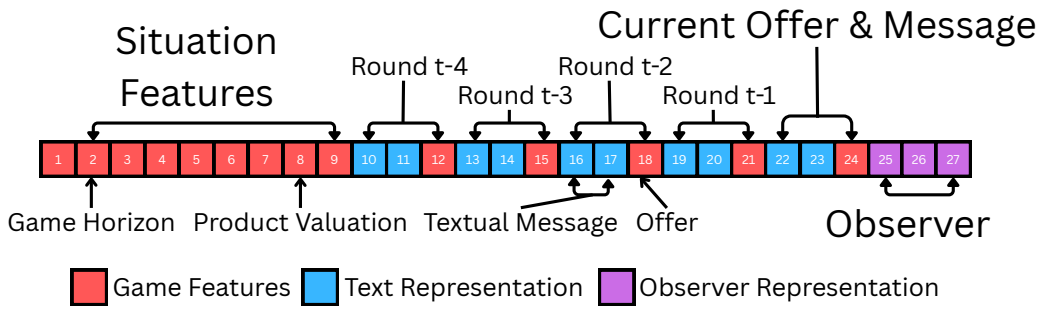


Figure 4: Schematic of one multimodal row. Structured fields (red) contain the recorded configuration, current offer, and fixed-width history; the generic text block (blue) encodes current-round messages; and the optional Observer block (purple) represents the reconstructed decision-time prompt. Cell counts are illustrative.

- *Discounting* (2)*:* `inflation_loss_1` and `inflation_loss_2`, the cumulative discounting incurred by each player up to round r (e.g., $1-\delta_1^{r-1}$ for player 1), again populated only when that quantity is visible to the deciding player.
- *History* (10)*:* for $h=1,\ldots,5$, `prev`h`_offer_frac` and `prev`h`_decision`: the proposer's split and responder's accept/reject at round $r-h$.
- *Family indicator* (1)*:* `family_idx` (constant within bargaining; included for cross-family compatibility of the schema).

Game-state features – negotiation (25 columns). At each responder decision we extract the following fields from the deciding player's observable view. The deciding player's reservation value is retained; the opponent's value and its derived outside-option field are encoded as `NaN` under incomplete information.

- *Configuration* (8)*:* `round`, `max_rounds`, `round_frac`, `sv` (seller's reservation valuation when observable), `bv` (buyer's reservation valuation when observable), `product_price_order` (typical-price scale used to order configurations), `messages`, `complete_info`.
- *Outside-option references* (2)*:* `seller_outside` ($=$`sv` $\cdot$ `product_price_order`) and `buyer_outside` ($=$`bv` $\cdot$ `product_price_order`): the absolute payoffs of the outside option for each player.
- *Current offer* (4)*:* `price`, `offer_frac`, `offer_vs_buyer_outside`, and `rounds_remaining`. These encode the proposed price, its ratio to the typical-price scale, its ratio to the buyer's outside option, and the remaining horizon.
- *History* (10)*:* `prev`h`_offer_frac` and `prev`h`_decision` for $h=1,\ldots,5$.
- *Family indicator* (1)*:* `family_idx`.

Dialogue representation. For each responder decision at round r, we collect every `message` field exchanged within round r in the game log, concatenate them with single-space separators, and encode the string with `sentence-transformers/all-MiniLM-L6-v2`, yielding a 384-dimensional vector. When the round contains no messages we use `"Round` r`"`. The vectors are PCA-reduced to 5 dimensions; the projection is fit on the conditioning rows of each evaluation cell and applied unchanged to test rows. We keep the same text dimensionality for baseline and Observer variants.

Agent-identity indicator. Let $\mathcal{S}$ denote the source agents in the conditioning set and t the held-out target. The identity indicator is one-hot over $\mathcal{S} \cup \{t\}$. At $K=0$, the target column is zero in all conditioning rows and active in test rows, so it adds no within-test signal. At $K>0$, rows from the target's adaptation games activate the same column and provide a within-target anchor.

PCA fitting. All PCA projections are fit on conditioning rows within each evaluation cell and applied unchanged to test rows. No test information enters PCA fitting.

O THINKING-BUDGET PILOT FOR THE LLM-AS-PREDICTOR BASELINE

The main LLM-as-Predictor table uses a full run with `thinking_budget=0`: the unified 72 bargaining and 39 negotiation response target pools, $K \in \{0, 2, 4, 8, 16\}$, and up to 30 test decisions per cell, matching the seed protocol of Section 5. To check whether this underpowers the direct LLM-as-Predictor baseline, we ran a stratified pilot on 5 target agents per family with `thinking_budget=2000`. The pilot cost was approximately \$31 for 4,010 API calls, with thinking tokens accounting for most of the cost; scaling the same configuration to the full table was estimated at roughly \$235, so we did not run it as a main experiment.

Table 11: Thinking-budget pilot on the same 5 target agents per family and matched cells. The rightmost column reports our cross-population response-prediction model (game+text features + Observer hidden-state representation) restricted to the pilot cells.

Family	K	LLM predictor (0)	LLM predictor (2000)	Observer model
Barg	0	.628	.656	.672
Barg	8	.792	.860	.857
Barg	16	.733	.806	**.881**
Nego	0	.492	**.700**	.623
Nego	8	.645	.636	**.771**
Nego	16	.720	.642	**.750**

The result is mixed rather than a monotone strengthening of the LLM-as-Predictor. Thinking improves some low-K cells, including negotiation at $K{=}0$, but hurts negotiation at high K on this subset. The pilot does not overturn the main comparison: a stronger direct LLM-as-Predictor can improve isolated cells, especially at low K, but the Observer-augmented model remains stronger at $K{=}16$ in both families.