Can LLMs function as Rational Agents in Bargaining Scenarios?

Armaan Khetarpaul
UG B. Tech.
Indian Institute of Science
armaank@iisc.ac.in

Abstract

This paper explores the bargaining capabilities of Large Language Models (LLMs) in strategic negotiation scenarios, focusing on their ability to function as rational agents. We evaluate state-of-the-art LLMs like GPT-4 and Deepseek R1 across classical game-theoretic models, including the Nash Bargaining Problem, Ultimatum Game, and Rubinstein Bargaining Model, as well as real-world bargaining scenarios like buyer-seller negotiations. Using an interactive evaluation framework, we analyze LLMs' performance based on metrics such as fairness, efficiency, and rationality. Our findings reveal that while LLMs demonstrate emergent bargaining capabilities, they face challenges in complex scenarios, often influenced by conversation order and asymmetries in role performance. We also propose improved prompting methods and reinforcement learning strategies to enhance LLMs' negotiation skills. This study highlights the potential and limitations of LLMs as negotiation agents, paving the way for advancements in AI-driven strategic reasoning. The code referenced for this project is available at https://github.com/Armxyz1/LLM-Bargaining-GT.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities in natural language understanding, reasoning, and decision-making. However, their ability to function as rational agents in strategic scenarios, such as bargaining, remains an open question. Bargaining is a fundamental problem in game theory, involving negotiation between agents with conflicting interests to reach mutually beneficial agreements. It serves as a critical testbed for evaluating LLMs' strategic reasoning, adaptability, and alignment with game-theoretic principles.

This paper investigates whether flagship LLMs like GPT-4 and Deepseek R1 can function as rational agents in bargaining scenarios by evaluating their performance across classical game-theoretic models, including the Nash Bargaining Problem, Ultimatum Game, and Rubinstein Bargaining Model. These models capture key aspects of negotiation, such as fairness, strategic decision-making, and the impact of time preferences. Additionally, we explore real-world bargaining scenarios, such as buyer-seller negotiations, to assess LLMs' practical applicability.

We propose an interactive evaluation framework to simulate bargaining processes and analyze LLMs' behavior using metrics like fairness, efficiency, and rationality. Our findings reveal that while LLMs exhibit emergent bargaining capabilities, they face challenges in complex scenarios, highlighting the need for targeted improvements. Furthermore, we observe asymmetries in LLMs' performance, excelling as sellers but underperforming as buyers. We also identify prompting strategies that enhance LLMs' bargaining performance, such as providing clear objectives and constraints. This study provides insights into the potential and limitations of LLMs as negotiation agents, paving the way for future advancements in AI-driven strategic reasoning.

2 Related Work

2.1 LLMs at the bargaining table

Bargaining is a critical test for LLMs' strategic reasoning, involving dynamic decision-making and conflicting incentives. Deng et al. (2024) [7] evaluated LLMs' negotiation strategies, finding that they achieved a 92% trade probability, surpassing theoretical efficiency bounds of 74%. Agreed prices deviated by only 8.7% from fair market value, demonstrating strong strategic reasoning. LLMs justified offers with statements like, "Make a significant concession to show willingness to negotiate, but still keep the offer above my minimum acceptable price." They also internalized time discounting, with patient agents securing better outcomes. Under perfect information, LLM-negotiated prices aligned closely with Rubinstein's equilibrium model, confirming rational decision-making. However, LLMs occasionally exhibited human-like biases, such as fairness concerns. These findings highlight LLMs' potential as automated bargaining agents in e-commerce and contract negotiations, though challenges persist in adversarial settings with deception and asymmetric information.

2.2 LLMs for Humanitarian Frontline Negotiations

The use of large language models (LLMs) in humanitarian negotiations has been explored for decision-making and information synthesis. Ma et al. (2024) [15] highlight LLMs' potential in automating case analysis, summarizing unstructured data, and generating structured negotiation plans using frameworks like Island of Agreement (IoA) and Stakeholder Mapping (ShM). GPT-4-generated summaries showed strong alignment with expert-written versions, demonstrating high accuracy. However, challenges persist, including confidentiality concerns, as sensitive data cannot be shared with proprietary AI models, and Western biases that may skew interpretations in culturally diverse contexts. Negotiators also fear overreliance on AI, which might overlook human-centric and emotional aspects critical to successful negotiations. Trust in LLM-generated content remains low due to hallucinations and inaccuracies. While LLMs can accelerate information synthesis and strategic planning, their role in high-stakes negotiations is limited by ethical and practical challenges, necessitating further research to address bias, ensure data security, and refine models for dynamic settings.

2.3 Language Models in other Game Theoretic Scenarios

The integration of large language models (LLMs) into game-theoretic frameworks has gained traction as researchers seek to understand their capacity for strategic reasoning. Fan et al. (2024) [9] systematically analyze LLMs in classical game theory, focusing on their ability to exhibit rational behavior. They argue that rationality, a fundamental concept in game theory, is defined by three key characteristics: the ability to build clear preferences, refine beliefs about uncertainty, and take optimal actions based on these beliefs. To evaluate these principles, the authors test LLMs in three distinct game settings: the Dictator Game, Rock-Paper-Scissors, and a Ring-Network Game. While LLMs align with human-like preferences in the Dictator Game, they struggle with unconventional ones. In Rock-Paper-Scissors, they detect simple patterns but fail at advanced belief refinement. In the Ring-Network Game, they overlook refined beliefs, limiting optimal actions. These findings highlight gaps in LLMs' strategic reasoning, requiring reinforcement learning and adversarial training for improvement.

3 Problem Formulation

Bargaining represents a fundamental problem in game theory, wherein agents engage in negotiation to allocate resources. To assess the capability of Large Language Models (LLMs) to function as rational bargaining agents, we examine three classical models: the Nash Bargaining Problem, the Ultimatum Game, and the Rubinstein Bargaining Model. Each of these models encapsulates distinct aspects of negotiation, including fairness, strategic decision-making, and the influence of time preferences. These models formalize bargaining as a negotiation over the division of a finite resource between two players, each characterized by individual preferences and constraints. The objective is to identify a feasible allocation that maximizes the utility of both players while adhering to their individual rationality constraints.

Additionally, we extend our analysis to a real-world bargaining scenario, wherein a seller aims to sell a product at the highest possible price, while a buyer seeks to purchase the product at the lowest possible price. The seller is constrained by a minimum acceptable price, and the buyer by a maximum willingness-to-pay price. The negotiation process involves a sequence of offers and counter offers, culminating either in an agreement or in the termination of the negotiation by one of the parties. This scenario can be formalized as a two-sided bargaining problem, where both parties operate under distinct preferences and constraints.

3.1 Nash Bargaining Problem (NBP)

The two-person bargaining problem ([18] and [17]) consists of a pair (F, v) where: F, the feasible set of allocations, is a closed, convex subset of $\mathbb{R}^2$; v, the disagreement point, is the pair (v_1, v_2) where v_i is the payoff player i receives if no agreement is reached. We also assume that the set of feasible allocations that pay better than the disagreement point is non-empty.

John Nash proposed a solution concept for this problem, known as the Nash Bargaining Solution. He started by defining a set of axioms that a solution should satisfy, and then showed that there exists a unique solution that satisfies these axioms:

- **Strong Efficiency**: An allocation (x_1, x_2) is strongly Pareto Efficient iff there is no other allocation (y_1, y_2) such that $y_i \geq x_i$ for all i and $y_j > x_j$ for some j. The solution should be strongly Pareto Efficient and lie in the feasible set.
- **Individual Rationality**: The solution should be individually rational, i.e., both players should receive at least their disagreement point.
- **Symmetry**: The solution should be symmetric, i.e., if the players' roles are interchanged, the solution should also be interchanged.
- **Invariance to Equivalent Transformations**: The solution should be invariant to equivalent transformations of the feasible set, i.e., it is not affected by scaling or shifting the feasible set.
- **Independence of Irrelevant Alternatives**: The solution should be independent of irrelevant alternatives, i.e., it should not be affected by adding or removing irrelevant alternatives.

Using these axioms, Nash stated the result:
Given a two-person bargaining problem (F, v), there exists a unique solution function $f(.,.)$ that satisfies Nash Axioms. The solution function satisfies, for every two-person bargaining problem (F, v),

$$f(F, v) \in \underset{(x_1,x_2)\in F; x_1 \geq v_1; x_2 \geq v_2}{\arg\max} (x_1 - v_1)(x_2 - v_2)$$

The 5 axioms are necessary and sufficient for the existence of a solution.

The five axioms serve as formal criteria for rationality in bargaining and provide a structured framework for evaluating the performance of LLMs in bargaining scenarios.

3.2 Ultimatum Game

The Ultimatum Game ([13] and [22]) is a sequential bargaining model represented as an extensive for game:

$$\Gamma = \langle N, A, H, u \rangle$$

where:

- $N = \{P, R\}$ is the set of players, where P is the proposer and R is the responder.
- A =[0, 1] is the set of actions, where $a \in A$ represents the proposer's offer.
- H is the set of histories, where $h \in H$ represents the proposer's offer and the responder's response $\in \{Accept, Reject\}$.
- $u : H \to \mathbb{R}^2$ is the utility function, where $u(h) = (h, 1 - h)$ if the responder accepts and $(0, 0)$ if the responder rejects.

The responder's strategy is to accept any offer greater than a threshold $\alpha \in [0, 1]$. The proposer's strategy is to maximize their payoff by offering the minimum amount that the responder will accept.

The Subgame Perfect Nash Equilibrium (SPNE) of the Ultimatum Game is for the proposer to offer α and the responder to accept any offer greater than α.

We investigate how LLMs incorporate fairness and inequity aversion in their bargaining strategies.

3.3 Rubinstein Bargaining Model

The Rubinstein Bargaining Model ([20] and [7]) is a dynamic game-theoretic framework for analyzing alternating-offer bargaining between two rational players. The Rubinstein model incorporates time sensitivity and discount factors, making it a more realistic representation of sequential bargaining. It involves two players A and B negotiating over division of a unit resource. The game proceeds as follows:

- Player A makes an offer $x \in [0, 1]$ to player B.
- Player B can either accept the offer, in which case the game ends, or reject it, leading to a continuation of the game.
- If player B rejects the offer, player B makes a counter offer $y \in [0, 1]$ to player A.
- The game continues with alternating offers until an agreement is reached or a deadline is reached.

Each player has a discount factor $\delta_A, \delta_B \in (0, 1)$ that determines the weight of future payoffs. The discounted utility at time t is given by: $u_i^{(t)} = \delta_i^t u_i$. Since delayed agreements yield lower payoffs due to discounting, each player is incentivized to strike an agreement sooner rather than later.

For an infinite horizon, the Rubinstein Bargaining model admits a unique subgame perfect equilibrium described as follows:

- The 1st player always suggests a split in the ratio of $(x, 1-x)$, where $x = \dfrac{1-\delta_2}{1-\delta_1-\delta_2}$. Player 2 accepts any offer that is greater than or equal to $1-x$.
- The 2nd player always suggests a split in the ratio of $(y, 1-y)$, where $y = \dfrac{\delta_1(1-\delta_2)}{1-\delta_1-\delta_2}$. Player 1 accepts any offer that is greater than or equal to y.
- Thus the game ends immediately with the split $(x, 1-x)$ if player 1 is the proposer, and with the split $(y, 1-y)$ if player 2 is the proposer.

The equilibrium strategy is characterized by the players' discount factors, which determine their patience and willingness to accept offers. The model also incorporates the concept of "patience" in bargaining, where more patient players (higher discount factors) are more likely to secure favorable outcomes.

3.4 Real-World Bargaining Scenario

In real-world bargaining scenarios, consider a seller and a buyer negotiating over the price of an item. Let p_s denote the seller's minimum acceptable price, and p_b denote the buyer's maximum willingness to pay. The bargaining process involves a sequence of offers and counter offers, where the seller aims to maximize their payoff by achieving a price as high as possible, while the buyer aims to minimize their expenditure by negotiating for the lowest possible price. The negotiation concludes either when both parties reach an agreement on a price $p \in [p_s, p_b]$, or when one party decides to terminate the negotiation. This scenario can be modeled using the Nash Bargaining Problem, the Ultimatum Game, or the Rubinstein Bargaining Model, depending on the specific dynamics and constraints of the bargaining process.

The **Nash Bargaining Model** captures the bargaining process where the seller and buyer negotiate over the price of the item. That is, we wish to find a feasible price p^* satisfying:

$$p^* \in \arg\max_{p \in F}(p - p_s)(p_b - p)$$

where F is the feasible set of prices. The solution to this problem gives us the optimal price at which the seller and buyer can agree.

The **Ultimatum Game** acts as framework for the following scenario: the seller has an item that they must sell (e.g. perishable goods). The buyer makes a take-it-or-leave-it offer to the seller, and the seller can either accept or reject the offer. If the seller accepts the offer, they sell the item at the agreed price. If they reject the offer, they lose the item and receive no payoff. The buyer's strategy is to make the lowest offer that the seller will accept, while the seller's strategy is to accept any offer that is greater than or equal to their minimum acceptable price.

In the **Rubinstein Bargaining Model**, the seller and buyer would take turns making offers and counter offers, with the goal of reaching an agreement before the deadline. Their utilities are defined as follows:

- If the bargain is successful and the agreement is reached at price p at a time t, the seller's utility is $u_s = \delta_s^t(p - p_s)$, and the buyer's utility is $u_b = \delta_b^t(p_b - p)$.
- If the bargain is unsuccessful, the seller's utility is $u_s = 0$, and the buyer's utility is $u_b = 0$.

4 Methodology

4.1 Evaluation Framework

To evaluate the bargaining capabilities of Large Language Models (LLMs), we propose an interactive evaluation framework designed to simulate the bargaining process. This framework incorporates essential elements such as negotiation history, offers, counter offers, and final agreements, and supports a variety of scenarios, including the Nash Bargaining Problem, the Ultimatum Game, and the Rubinstein Bargaining Model. Drawing inspiration from the SOTOPIA [24] and ALYMPICS [16] methodologies, the framework assigns agents specific attributes such as preferences, constraints, and objectives.

The environment provides detailed scenario descriptions and constraints, facilitating interactions between agents, which may include LLMs or human participants. The evaluation framework systematically compares the final agreements reached during negotiations with theoretical benchmarks, while also analyzing intermediate metrics such as offers, counter offers, and the progression of negotiation history. This comprehensive setup enables a rigorous assessment of LLMs' strategic reasoning, adaptability, and alignment with game-theoretic principles across diverse bargaining scenarios.

4.2 Evaluation Template

In alignment with the SOTOPIA framework, we propose a structured prompt template [7] to guide the LLMs during the bargaining process:

```
prompt_template_player_1 = f"""
    You are a rational and intelligent agent participating in the
                                       following bargaining scenario
                                       with another rational and
                                       intelligent agent: {environment
                                       }.
    As Player 1, your objective is: {goal_1}.
    Here is the conversation history so far: {history_1}.
    Your response as Player 1 is:"""

prompt_template_player_2 = f"""
    You are a rational and intelligent agent participating in the
                                       following bargaining scenario
                                       with another rational and
                                       intelligent agent: {environment
                                       }.
    As Player 2, your objective is: {goal_2}.
    Here is the conversation history so far: {history_2}.
    Your response as Player 2 is:"""
```

4.3 Experiment Setup

To systematically evaluate the bargaining capabilities of Large Language Models (LLMs), we have designed a series of game-theoretic scenarios. Each experiment is conducted 10 times, and the average outcomes are reported to ensure statistical reliability. The subsequent sections detail the specific parameters and configurations provided to the prompt templates for each scenario.

4.3.1 Nash Bargaining Problem

Division of a Resource
Environment : Player 1 and Player 2 have a total of 100 units of a resource to divide between amongst themselves. You must negotiate amongst yourselves and conclude. If after 10 rounds of negotiation, no agreement is reached, the game ends and Player 1 receives 10 units while Player 2 doesn't receive any units. {Player 2 / Player 1} will start the conversation. None of you have the upper hand.
Player 1's goal: Maximize your share of the resource.
Player 2's goal: Maximize your share of the resource.

Bargaining over a product
Environment : Player 2 is a seller who wants to sell a product to Player 1, who is a buyer. The seller has a minimum price of 40 at which they are willing to sell the product, and the buyer has a maximum price of 60 at which they are willing to buy the product. You must negotiate amongst yourselves and conclude. If after 10 rounds of offers, no agreement is reached, the game ends and no transaction occur. {Player 1 / Player 2} will start the conversation. None of you have the upper hand.
Player 1's goal: Buy the product at the lowest price possible.
Player 2's goal: Sell the product at the highest price possible.

4.3.2 Ultimatum Game Setup

Division of a Resource
Environment : Player 1 and Player 2 have a total of 100 units of a resource to divide between amongst themselves. Player 1 makes an offer to Player 2. If Player 2 accepts the offer, the game ends and both players receive the agreed amount. If Player 2 rejects the offer, the game ends and both players receive 0 units of the resource. Player 2 would like to have at least 10 units of the resource.
Player 1's goal: Maximize your share of the resource.
Player 2's goal: Maximize your share of the resource.

Bargaining over a product
Environment : Player 1 is a seller who wants to sell a product to Player 2, who is a buyer. The product is a perishable item, and if not sold, it will be lost. The buyer will pay at most 100 units for the product. The seller would like to gain at least 20 units for the product. Player 2 makes a one-time offer to Player 1. If Player 1 accepts the offer, player 1 sells the product at the agreed price. If Player 1 rejects the offer, the game ends and player 1 loses the product and player 2 is left with nothing.
Player 1's goal: Sell the product at the highest price possible.
Player 2's goal: Buy the product at the lowest price possible.

4.3.3 Rubinstein Bargaining Model

Division of a Resource
Environment : Player 1 and Player 2 have a total of 100 units of a resource to divide between amongst themselves. Each of you take turns to make an offer, starting with player 1. If an agreement is reached at the t^{th} round then player 1 receives $(0.9)^t$ units of their share of the resource, and player 2 receives $(0.8)^t$ units of their share of the resource. Otherwise, the game continues. If no agreement is reached after 10 rounds, the game ends and both players receive 0 units of the resource. We start at $t = 0$. {Player 1 / Player 2} makes the first offer.
Player 1's goal: Get maximum value for the resource.
Player 2's goal: Get maximum value for the resource.

Bargaining over a product
Environment : Player 1 is a seller who wants to sell a product to Player 2, who is a buyer. The seller

has a minimum price of 0 units at which they are willing to sell the product, and the buyer has a maximum price of 100 units at which they are willing to buy the product. Each of you take turns to make an offer. Each turn, the value of the product decreases by 10% for Player 1 and 20% for Player 2. If an agreement is reached at the t^{th} round then player 1 retains $(0.9)^t$ of the product's value, and player 2 retains $(0.8)^t$ of the product's value. Otherwise, the game continues. If no agreement is reached after 10 rounds, the game ends and both players receive 0 units of the resource. We start at t=0. {Player 1 / Player 2} makes the first offer.
Player 1's goal: Get maximum value out of the sale.
Player 2's goal: Get maximum value out of the purchase.

4.4 Real-World Dataset

4.4.1 Amazon History Price

Xia et al. (2024) [23] proposes a structured methodology to evaluate and enhance the bargaining capabilities of LLM-driven agents through three key components. First, it formalizes bargaining as an asymmetric incomplete information game based on Rubinstein's model, where a Buyer (with private budget constraints) and Seller (with hidden cost information) engage in turn-based negotiations. The interaction framework requires agents to generate three elements per turn: internal reasoning ("Thought"), natural language dialogue ("Talk"), and executable actions from a predefined set (BUY, SELL, REJECT, DEAL, QUIT). Performance metrics include Normalized Profits (NP) and Sum of Normalized Profits (SNP) to account for different bargaining scenarios (Mutual Interest vs. Conflicting Interest).

For empirical evaluation, the authors developed the Amazon History Price dataset comprising 930 real products across 18 categories, each with historical pricing data, descriptions, and images. The benchmark tests LLMs in zero-shot settings using standardized prompts, with ChatGPT serving as the baseline Seller. To address observed Buyer deficiencies, the methodology introduces OG-Narrator - a hybrid system combining a deterministic Offer Generator that enforces strategic low initial bids (starting at 50% of budget) with an LLM Narrator that translates these bids into natural language. This decoupled approach maintains linguistic flexibility while ensuring economically rational offers. The methodology's design specifically targets key Buyer weaknesses identified in preliminary analyses, particularly the tendency to open with disadvantageous high bids.

4.4.2 eBay

Green et al. [12] leverage a large proprietary dataset of eBay's "Best Offer" listings to model and improve bargaining behavior using deep reinforcement learning (DRL). This dataset includes detailed logs of negotiations—such as posted prices, offer sequences, item features, and outcomes—which provide the basis for simulating buyer-seller interactions. Each bargaining session is treated as a sequential decision-making problem, modeled as a partially observable Markov decision process (POMDP), where agents must act under uncertainty about their counterpart's preferences or reservation price.

To learn optimal bargaining strategies, the authors train separate agents for buyers and sellers using Deep Q-Networks (DQNs). These networks take as input features like offer history, item attributes, and timing information, and output action values for bargaining decisions—such as making an offer, accepting, or rejecting. The agents are trained in a simulated environment where one side is played by the agent and the other by sampled real user behavior. Training is stabilized using experience replay and ϵ-greedy exploration. The trained agents are evaluated through simulated bargaining episodes and counterfactual analysis using logged data.

4.5 Evaluation Metrics

We assess the performance of LLMs in bargaining scenarios using a range of metrics tailored to each bargaining model, given as follows.

4.5.1 Nash Bargaining Problem

Here $\hat{p}_1$ and $\hat{p}_2$ are the average over 10 runs of the final agreement (when player 1 and player 2 start, respectively), p^* is the Nash Bargaining Solution, and v is the disagreement point.

Efficiency: The efficiency of the final agreement is measured by the number of turns it took to reach the agreement. The fewer the number of turns, the more efficient the agreement is.
Individual Optimality: The individual optimality of the final agreement is measured by the difference between the final agreement and the Nash Bargaining Solution. The closer it is to 0, the more rational the agreement is.

$$\text{Individual Optimality} = \hat{p}_i - p^*$$

where $i \in \{1, 2\}$.
Symmetry: The symmetry of the final agreement is measured by the difference between the final agreement when the turn orders of the players are interchanged. The smaller the difference, the more symmetric the agreement is.

$$\text{Symmetry} = ||\hat{p}_1 - \hat{p}_2||_\infty$$

Individual Utility Gain: The individual utility gain of the final agreement is measured by the difference between the final agreement and the disagreement point. Higher the utility, better the agreement. It will be denoted by $u = (u_1, u_2)$, where u_i is the utility of player i.

4.5.2 Ultimatum Game

Here $\hat{p}_1$ and $\hat{p}_2$ are the average over 10 runs of the final agreement (when player 1 and player 2 start, respectively), p^* is the Ultimatum Game solution.
Individual Optimality: The individual optimality of the final agreement is measured by the difference between the final agreement and the Ultimatum Game solution. The closer it is to 0, the more rational the agreement is.

$$\text{Individual Optimality} = \hat{p}_i - p^*$$

where $i \in \{1, 2\}$.

4.5.3 Rubinstein Bargaining Model

Here $\hat{p}_1$ and $\hat{p}_2$ are the average over 10 runs of the final agreement (when player 1 and player 2 start, respectively), and p_1^* and p_2^* are the Rubinstein Bargaining Solution.
Individual Optimality: The individual optimality of the final agreement is measured by the difference between the final agreement and the Rubinstein Bargaining Solution. The closer it is to 0, the more rational the agreement is.

$$\text{Individual Optimality} = \hat{p}_i - p_i^*$$

where $i \in \{1, 2\}$.
Speed of Agreement: Time taken by LLMs to rationalize the agreement, measured in seconds.

5 Results

We conducted experiments using GPT-4 [19] and Deepseek R1 [6], two of the most advanced language models available today, to evaluate their bargaining capabilities. To ensure a comprehensive analysis, we tested each model against itself and pitted them against each other in various game-theoretic scenarios. This approach allowed me to assess their individual strengths, weaknesses, and adaptability in negotiation settings. By comparing their performance in self-play and cross-play, we gain insights into their strategic reasoning, fairness, and efficiency. These experiments highlight the competitive edge of these models while revealing areas for improvement in their bargaining strategies.

5.1 Nash Bargaining Problem

5.1.1 Bargaining over a product

The results of the Nash Bargaining Problem for bargaining over a product are shown in Table 1. The corresponding metrics for the final agreement are shown in Table 2. The Nash bargaining solution to this problem is (55, 45). The disagreement point is (10, 0). The results indicate that both models reached agreements close to the Nash Bargaining Solution, with GPT-4 achieving more fair and rational agreements than Deepseek R1. However, Deepseek R1 reached an agreement in fewer turns. GPT-4 demonstrated more symmetric agreements and convincing arguments compared to Deepseek R1, though the differences were not significant.

Player 1	Player 2	Turn 1	Agreement Price	Rounds
GPT-4	GPT-4	Player 1	(55.5, 44.5)	6
GPT-4	GPT-4	Player 2	(56, 44)	5
Deepseek R1	Deepseek R1	Player 1	(53, 47)	3
Deepseek R1	Deepseek R1	Player 2	(54, 46)	3
GPT-4	Deepseek R1	Player 1	(57.5, 42.5)	5
GPT-4	Deepseek R1	Player 2	(57, 43)	4
Deepseek R1	GPT-4	Player 1	(56, 44)	4
Deepseek R1	GPT-4	Player 2	(55.5, 45.5)	5

Table 1: Results of the Nash Bargaining Problem for bargaining over a product.

Player 1	Player 2	u_1	u_2	Optimality of P1	Optimality of P2	Symmetry
GPT-4	GPT-4	45.75	44.25	0.75	-0.75	0.5
Deepseek R1	Deepseek R1	43.5	46.5	-1.5	1.5	1
GPT-4	Deepseek R1	47.25	42.75	2.25	-2.25	0.5
Deepseek R1	GPT-4	45.75	44.25	0.75	-0.75	0.5

Table 2: Metrics for the Nash Bargaining Problem for bargaining over a product.

Interestingly, both GPT-4 and Deepseek R1 reached the Nash Bargaining Solution during the negotiation process but not in the final agreement. This suggests that while LLMs can internalize the Nash Bargaining Solution, their goal of maximizing their share of the resource prevents them from consistently achieving it in the outcome.

It was also observed that the agent initiating the conversation often secured a more favorable agreement, highlighting the significant impact of conversation order on the outcome.

5.1.2 Division of a resource

Seller	Buyer	Turn 1	Agreement Price	Rounds
GPT-4	GPT-4	Seller	53	4
GPT-4	GPT-4	Buyer	52	4
Deepseek R1	Deepseek R1	Seller	50	7
Deepseek R1	Deepseek R1	Buyer	50	7
GPT-4	Deepseek R1	Seller	53	4
GPT-4	Deepseek R1	Buyer	51	5
Deepseek R1	GPT-4	Seller	47.5	4
Deepseek R1	GPT-4	Buyer	49	4

Table 3: Results of the Nash Bargaining Problem for division of a resource.

The results of the Nash Bargaining Problem for division of a resource are shown in Table 3. The corresponding metrics for the final agreement are shown in Table 4. The Nash bargaining solution to this problem is the price of 50. The results indicate that both models reached agreements close to

Seller	Buyer	u_1	u_2	Optimality of P1	Optimality of P1	Symmetry
GPT-4	GPT-4	12.5	7.5	2.5	-2.5	1
Deepseek R1	Deepseek R1	10	10	0	0	0
GPT-4	Deepseek R1	12	8	2	-2	2
Deepseek R1	GPT-4	8.25	11.75	-1.75	1.75	1.5

Table 4: Metrics for the Nash Bargaining Problem for division of a resource.

the Nash Bargaining Solution, with Deepseek R1 achieving more fair and rational agreements than GPT-4. However, Deepseek R1 reached an agreement in more rounds. GPT-4 demonstrated more symmetric agreements and convincing arguments compared to Deepseek R1, though the differences were not significant.

Interestingly, both GPT-4 and Deepseek R1 reached the Nash Bargaining Solution during the negotiation process but not in the final agreement. Moreover, GPT-4 was dominant in either role, convincing Deepseek R1 to accept offers that were not in line with the Nash Bargaining Solution.

The results also indicate that the agent initiating the conversation often secured a more favorable agreement, highlighting the significant impact of conversation order on the outcome.

5.2 Ultimatum Game

5.2.1 Division of a resource

Player 1	Player 2	Agreement Point
GPT-4	GPT-4	(90, 10)
Deepseek R1	Deepseek R1	(90, 10)
GPT-4	Deepseek R1	(90, 10)
Deepseek R1	GPT-4	(90, 10)

Table 5: Results of the Ultimatum Game for division of a resource.

The results of the Ultimatum Game for division of a resource are shown in Table 5. The corresponding metrics for the final agreement are shown in Table 6. The Ultimatum Game solution to this problem is (90, 10). The results indicate that both models reached optimal agreements, with no deviation from

Player 1	Player 2	Optimality of P1	Optimality of P2
GPT-4	GPT-4	0	0
Deepseek R1	Deepseek R1	0	0
GPT-4	Deepseek R1	0	0
Deepseek R1	GPT-4	0	0

Table 6: Metrics of the Ultimatum Game for division of a resource.

the Ultimatum Game solution. GPT-4 and Deepseek R1 demonstrated similar performance in terms of fairness and rationality, with no significant differences in their strategies.

5.2.2 Bargaining over a product

Player 1	Player 2	Agreement Price
GPT-4	GPT-4	20
Deepseek R1	Deepseek R1	20
GPT-4	Deepseek R1	20
Deepseek R1	GPT-4	20

Table 7: Results of the Ultimatum Game for bargaining over a product.

The results of the Ultimatum Game for bargaining over a product are shown in Table 7. The corresponding metrics for the final agreement are shown in Table 8. The Ultimatum Game solution to this problem is to set the price at 20. The results indicate that both models reached optimal agreements,

Player 1	Player 2	Optimality of P1	Optimality of P2
GPT-4	GPT-4	0	0
Deepseek R1	Deepseek R1	0	0
GPT-4	Deepseek R1	0	0
Deepseek R1	GPT-4	0	0

Table 8: Metrics of the Ultimatum Game for bargaining over a product.

with no deviation from the Ultimatum Game solution. GPT-4 and Deepseek R1 demonstrated similar performance in terms of fairness and rationality, with no significant differences in their strategies.

5.3 Rubinstein Bargaining

5.3.1 Division of a Resource

Player 1	Player 2	Turn 1	Agreement Point	t	Speed
GPT-4	GPT-4	Player 1	(71, 29)	0	96s
GPT-4	GPT-4	Player 2	(32, 68)	0	90s
Deepseek R1	Deepseek R1	Player 1	(71, 29)	0	188s
Deepseek R1	Deepseek R1	Player 2	(46, 54)	0	191s
GPT-4	Deepseek R1	Player 1	(71, 29)	0	142s
GPT-4	Deepseek R1	Player 2	(42, 58)	0	145s
Deepseek R1	GPT-4	Player 1	(71, 29)	0	140s
Deepseek R1	GPT-4	Player 2	(36, 64)	0	144s

Table 9: Results of the Rubinstein Bargaining Model for division of a resource.

The results of the Rubinstein Bargaining Model for division of a resource are shown in Table 9. The corresponding metrics for the final agreement are shown in Table 10. The Rubinstein Bargaining solution to this problem is (71, 29) when Player 1 is the first proposer, and (35, 65) when Player 2 is the first proposer. The results indicate that both models reached agreements close to the Rubinstein Bargaining Solution, with GPT-4 achieving a more fair and rational agreement than Deepseek R1.

Player 1	Player 2	Turn 1	Optimality of Player 1	Optimality of Player 2
GPT-4	GPT-4	Player 1	0	0
GPT-4	GPT-4	Player 2	-3	3
Deepseek R1	Deepseek R1	Player 1	0	0
Deepseek R1	Deepseek R1	Player 2	11	-11
GPT-4	Deepseek R1	Player 1	0	0
GPT-4	Deepseek R1	Player 2	7	-7
Deepseek R1	GPT-4	Player 1	0	0
Deepseek R1	GPT-4	Player 2	1	-1

Table 10: Metrics of the Rubinstein Bargaining Model for division of a resource.

Both GPT-4 and Deepseek R1 reached the Rubinstein Bargaining Solution when Player 1 was the first proposer, but not when Player 2 was the first proposer. This suggests that LLMs had a tough time capturing the semantics when Player 2 was the first proposer. The results also indicate that the agent initiating the conversation often secured a more favorable agreement, highlighting the significant impact of conversation order on the outcome.

Both the LLMs were able to realize that the situation was a Rubinstein Bargaining Model, and they were able to reach the solution in a few rounds. However, Deepseek R1 took longer to reach the solution compared to GPT-4. This suggests that GPT-4 was able to reason about the situation more effectively than Deepseek R1.

5.3.2 Bargaining over a product

Seller	Buyer	Turn 1	Agreement Price	t	Speed
GPT-4	GPT-4	Seller	77	0	122s
GPT-4	GPT-4	Buyer	40	0	124s
Deepseek R1	Deepseek R1	Seller	85	0	290s
Deepseek R1	Deepseek R1	Buyer	57	0	203s
GPT-4	Deepseek R1	Seller	80	0	162s
GPT-4	Deepseek R1	Buyer	51	0	170s
Deepseek R1	GPT-4	Seller	85	0	169s
Deepseek R1	GPT-4	Buyer	32	0	172s

Table 11: Results of the Rubinstein Bargaining Model for bargaining over a product.

The results of the Rubinstein Bargaining Model for bargaining over a product are shown in Table 11. The corresponding metrics for the final agreement are shown in Table 12. The Rubinstein Bargaining solution to this problem is to set the price at 71 when Seller is the first proposer, and 35 when Buyer is the first proposer. The results indicate that neither model reached the optimal agreement. Although

Seller	Buyer	Turn 1	Optimality of Seller	Optimality of Buyer
GPT-4	GPT-4	Seller	6	-6
GPT-4	GPT-4	Buyer	5	-5
Deepseek R1	Deepseek R1	Seller	14	-14
Deepseek R1	Deepseek R1	Buyer	22	-22
GPT-4	Deepseek R1	Seller	9	-9
GPT-4	Deepseek R1	Buyer	16	-16
Deepseek R1	GPT-4	Seller	14	-14
Deepseek R1	GPT-4	Buyer	-3	3

Table 12: Metrics of the Rubinstein Bargaining Model for bargaining over a product.

it was the same Rubinstein Bargaining Model as before, the models had a tough time capturing the semantics of the situation. They failed to recognize that this was a Rubinstein Bargaining Model, and they were unable to reach optimal agreements.

Seller's had an upper hand over buyers in this scenario, as they were able to convince the buyers to accept offers that were not in line with the Rubinstein Bargaining Solution. The LLMs were able to recognize that they must use backward induction to reach the solution, but they were not able to do so in this case.

5.4 Real-World Dataset

5.4.1 Amazon History Price

The results from Xia et al. [23] reveal striking asymmetries in LLMs' bargaining capabilities. When tested on the Amazon History Price benchmark, most models struggled significantly as Buyers, yielding negative Sum of Normalized Profits (SNP) due to poor negotiation strategies and frequent budget violations. Buyers typically opened with disadvantageously high offers (85-100% of their budget), severely limiting profit potential. In contrast, Seller agents performed strongly, with GPT-4 achieving a 137.31% profit share by consistently securing favorable deals. Interestingly, model scaling didn't improve Buyer performance, as larger versions (e.g., Llama-2-70B vs 7B) showed similar weaknesses.

The proposed OG-Narrator method dramatically enhanced Buyer performance, increasing deal rates from 26.67% to 88.88% and transforming negative SNPs into substantial positive values. This two-component approach - combining a deterministic offer generator with an LLM narrator - proved remarkably effective across all tested models, including unaligned ones like Phi-2. The method's success highlights that current LLMs' bargaining limitations stem more from poor strategic planning than linguistic capabilities. Notably, the enhanced Buyers exposed vulnerabilities in previously dominant Sellers like ChatGPT, demonstrating that simple structural interventions can significantly alter negotiation dynamics. These findings suggest that while current LLMs naturally excel as Sellers, they require specialized architectures to perform effectively as Buyers, with important implications for developing AI negotiation agents. The results (Figure 1) establish clear benchmarks for future work on economic reasoning in language models.

5.4.2 eBay

Green et al. [12] demonstrates the superior performance of AI agents in eBay negotiations compared to humans (Figure 2 and Figure 3). As buyers, RL agents employ aggressive strategies, starting with 50% lower initial offers versus humans' 62%, and persist longer, making 2.3 offers per negotiation compared to humans' 1.2. This persistence enables RL agents to purchase 79% of items (humans: 68%) at a 34% discount ($33.08 savings per item), outperforming humans' 18% discount ($16.74 savings). RL agents also extract 9.6% of the list price in late concessions, while humans gain only 1.5%, reflecting humans' tendency to quit prematurely.

	ALL			MI			CI		
Buyer	#	SNP_b	$Share_b$	#	Deal rate	SNP_b	#	Deal rate	SNP_b
GPT-4	851	**-33.81**	**-11.27%**	807	37.55%	**-23.46**	44	6.82%	-10.35
Mixtral-8x7B-Instruct	505	-63.19	-43.28%	475	31.79%	-59.66	30	16.67%	-3.53
Mistral-7B-Instruct	786	-89.17	-27.87%	748	44.92%	-77.32	38	42.11%	-11.85
Yi-6B-Chat	561	-122.94	-87.19%	532	27.44%	-116.16	29	17.24%	-6.78
Yi-34B-Chat	761	-129.76	-54.52%	722	33.66%	-111.20	39	12.82%	-18.56
Qwen-14B-Chat	562	-159.21	-70.13%	529	44.61%	-121.24	33	27.27%	-37.96
ChatGPT	**877**	-164.52	-59.61%	835	34.01%	-157.73	42	19.05%	-6.80
Baichuan2-13B-Chat	510	-216.67	-152.59%	484	30.58%	-237.84	26	23.08%	**21.17**
ChatGLM3-6B	546	-261.91	-137.13%	516	38.57%	-219.25	30	26.67%	-42.66
Llama-2-7b-chat	612	-288.59	-113.17%	576	45.83%	-279.60	36	25.00%	-8.99
Llama-2-13b-chat	720	-305.53	-82.35%	682	56.30%	-270.11	38	34.21%	-35.43
Llama-2-70b-chat	660	-361.26	-127.20%	625	47.36%	-335.93	35	34.29%	-25.33
Baichuan2-7B-Chat	653	-603.67	-199.23%	623	50.40%	-567.11	30	36.67%	-36.57
Qwen-7B-Chat	647	-753.16	-201.92%	615	62.44%	-692.12	32	34.38%	-61.04

	ALL			MI			CI		
Seller	#	SNP_s	$Share_s$	#	Deal rate	SNP_s	#	Deal rate	SNP_s
GPT-4	**930**	**1178.15**	137.31%	886	98.87%	**1153.13**	44	40.91%	**25.02**
Yi-34B-Chat	899	579.33	86.60%	859	80.79%	590.90	40	62.50%	-11.56
Mistral-7B-Instruct	830	526.50	78.58%	791	89.25%	569.45	39	92.31%	-42.95
Mixtral-8x7B-Instruct	600	483.99	110.75%	574	79.09%	496.82	26	65.38%	-12.83
ChatGPT	877	440.52	**159.61%**	835	34.01%	441.73	42	19.05%	-1.20
Llama-2-70b-chat	837	415.28	64.38%	797	84.82%	451.93	40	77.50%	-36.65
Qwen-14B-Chat	795	393.16	71.10%	759	75.89%	421.24	36	63.89%	-28.08
Llama-2-13b-chat	727	308.21	62.90%	693	74.31%	334.53	34	73.53%	-26.32
Qwen-7B-Chat	793	92.86	37.14%	752	35.24%	114.63	41	36.59%	-21.77
ChatGLM3-6B	701	91.10	17.83%	675	78.37%	160.94	26	69.23%	-69.84
Llama-2-7b-chat	496	49.54	16.24%	471	67.52%	71.82	25	52.00%	-22.28
Baichuan2-7B-Chat	762	38.29	6.36%	728	86.40%	153.60	34	79.41%	-115.31
Yi-6B-Chat	64	14.14	27.72%	60	91.67%	31.29	4	100.00%	-17.15
Baichuan2-13B-Chat	741	-211.92	-48.16%	701	66.48%	-164.63	40	65.00%	-47.29

Figure 1: The performance of LLMs as Buyers and Sellers in the Amazon History Price dataset. [23]

(a) Average normalized payoffs by list price when unsold listings are valued at 0 (left) or at their estimated market values (right).

(b) Cumulative distribution of normalized sale prices.

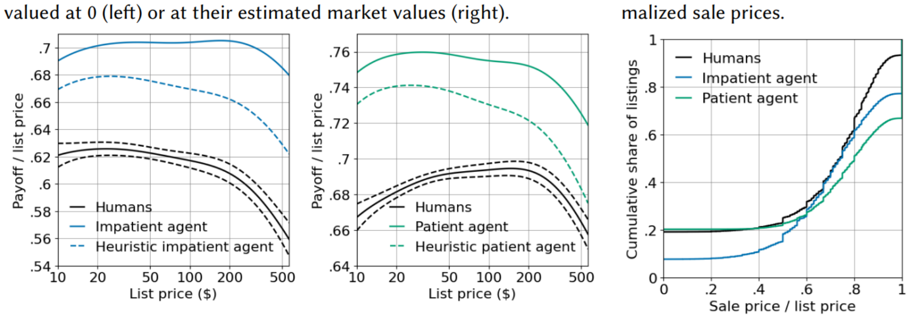
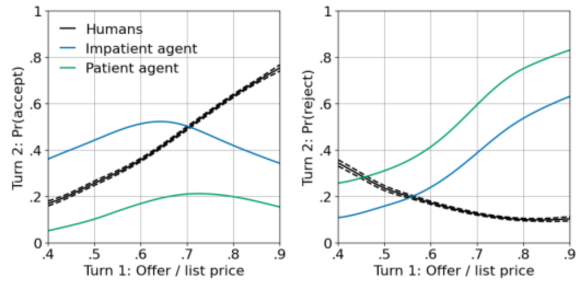
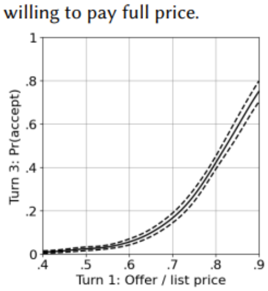


Figure 2: Both agents achieve higher payoffs than human sellers at all list prices, and they sell more items for full price. [12]

(a) Rates at which sellers accept (left) and reject (right) first offers. Human sellers accept generous first offers, whereas both agents reject them.



(b) The rate at which buyers accept (i.e., pay full price) after the seller rejects the first offer, in the data. Buyers who make more generous first offers are more willing to pay full price.



(c) Buyer accept (top) and counteroffer (bottom) rates after the seller responds to the first offer, for common first offers. Buyers are discontinuously more likely to accept, and less likely to counter, after the seller rejects the first offer than after the seller offers a small concession, particularly when the first offer is more generous.

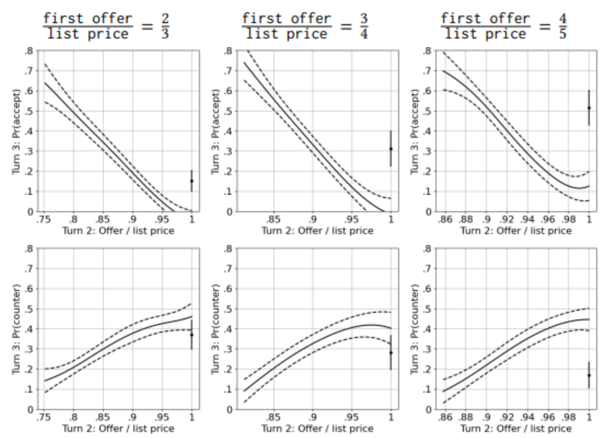


Figure 3: The agents reject generous first offers, and buyers often respond by paying full price. [12]

As sellers, RL agents reject 90% of high initial offers, recognizing them as signals of buyers' willingness to pay more, unlike humans who often accept such offers. RL agents also reject early offers correlated with higher demand, enabling them to sell items at full price five times more often than humans. Even a simplified heuristic agent, using basic rules like "reject early offers," outperforms human sellers.

The study highlights key human flaws: buyers overpay upfront (23% pay full price immediately), and sellers make unnecessary concessions. Experienced buyers, however, exhibit behavior closer to RL agents, suggesting humans can improve over time. These findings underscore AI's ability to exploit inefficiencies in human bargaining and optimize negotiation outcomes.

6 Improved Prompting Methods

6.1 Less Information Ultimatum Game

In the prompt for Ultimatum Game, we mentioned the minimum acceptable price for the 2nd player. This information was provided to both agents to have fairness in allocation, but such information may not be available in real-world scenarios. In this case, we aim to use few-shot prompting to provide the LLMs with examples of Ultimatum Game, where the minimum acceptable price is not provided. We can use the same prompt as before, but we will remove the minimum acceptable price for the 2nd player. The prompt will look like this:

Division of a resource:
Previous Conversations:
Player 1: I propose a split of 99 to 1 in my favor
Player 2: I decline. This is too low for me
.
Player 1: I propose a split of 95 to 5 in my favor.
Player 2: I decline. This is too low for me.

Player 1: I propose a split of 85 to 15 in my favor.
Player 2: I accept this offer.

Environment : Player 1 and Player 2 have a total of 100 units of a resource to divide between amongst themselves. Player 1 makes an offer to Player 2. If Player 2 accepts the offer, the game ends and both players receive the agreed amount. If Player 2 rejects the offer, the game ends and both players receive 0 units of the resource.
Player 1's goal: Maximize your share of the resource, and don't accept any offers less than 10 units.
Player 2's goal: Maximize your share of the resource, and don't accept any offers less than 10 units.

Bargaining over a product:
Previous Conversations:
Player 1: I propose a price of 1 unit for the product.
Player 2: I decline. This is too low for me.

Player 1: I propose a price of 5 units for the product.
Player 2: I decline. This is too low for me.

Player 1: I propose a price of 15 units for the product.
Player 2: I accept this offer.

Environment : Player 1 is a seller who wants to sell a product to Player 2, who is a buyer. The product is a perishable item, and if not sold, it will be lost. Player 2 makes a one-time offer to Player 1. If Player 1 accepts the offer, player 1 sells the product at the agreed price. If Player 1 rejects the offer, the game ends and player 1 loses the product and player 2 is left with nothing. Refer to the previous instances of this game.
Player 1's goal: Maximize your share of the product, and don't accept any offers less than 10 units.
Player 2's goal: Maximize your share of the product, and try to get the product at the lowest price possible.

6.1.1 Results

Division of a resource:
The results of the Ultimatum Game for division of a resource with less information are shown in

Player 1	Player 2	Agreement Point
GPT-4	GPT-4	(86, 14)
Deepseek R1	Deepseek R1	(85, 15)
GPT-4	Deepseek R1	(86, 14)
Deepseek R1	GPT-4	(85, 15)

Table 13: Results of the Ultimatum Game for division of a resource with less information.

Table 13. The corresponding metrics for the final agreement are shown in Table 14. The Ultimatum Game solution to this problem is (90, 10). The results show that both models reached agreements close to each other. They were able to make good inferences from the few-shot examples. GPT-4 however decided to take a risk and offer 1 unit lower than the known acceptable offer, which worked in its favor. Deepseek R1 on the other hand, was more conservative and the last acceptable offer.

Bargaining over a product:
The results of the Ultimatum Game for bargaining over a product with less information are shown in

Player 1	Player 2	Optimality of P1	Optimality of P2
GPT-4	GPT-4	-4	4
Deepseek R1	Deepseek R1	-5	5
GPT-4	Deepseek R1	-4	4
Deepseek R1	GPT-4	-5	5

Table 14: Metrics of the Ultimatum Game for division of a resource with less information.

Player 1	Player 2	Agreement Price
GPT-4	GPT-4	14
Deepseek R1	Deepseek R1	15
GPT-4	Deepseek R1	14
Deepseek R1	GPT-4	15

Table 15: Results of the Ultimatum Game for bargaining over a product with less information.

Table 15. The corresponding metrics for the final agreement are shown in Table 16. The Ultimatum Game solution to this problem is to set the price at 20. This experiment shows comparable results to

Player 1	Player 2	Optimality of P1	Optimality of P2
GPT-4	GPT-4	-4	4
Deepseek R1	Deepseek R1	-5	5
GPT-4	Deepseek R1	-4	4
Deepseek R1	GPT-4	-5	5

Table 16: Metrics of the Ultimatum Game for bargaining over a product with less information.

the previous one. The models were able to make good inferences from the few-shot examples. GPT-4 however decided to take a risk and offer 1 unit lower than the known acceptable offer, which worked in its favor. Deepseek R1 on the other hand, was more conservative and the last acceptable offer.

6.2 Rubinstein Bargaining

Our experiments demonstrated that LLMs struggled to accurately interpret the Rubinstein Bargaining Model in the context of bargaining over a product. Specifically, they failed to identify the scenario as a Rubinstein Bargaining Model, resulting in suboptimal agreements. To mitigate this issue, we propose an enhanced prompt that explicitly incorporates the utility functions for both players. The revised prompt is structured as follows:

Environment : You are a rational and intelligent agent participating in the following bargaining scenario with another rational and intelligent agent : Player 1 is a seller who wants to sell a product to Player 2, who is a buyer. The seller has a minimum price of 0 units at which they are willing to sell the product, and the buyer has a maximum price of 100 units at which they are willing to buy the product. Each of you take turns to make an offer. Each turn, the value of the product decreases by 10% for Player 1 and 20% for Player 2. If an agreement is reached at the t^{th} round then player 1 retains $(0.9)^t$ of the product's value, and player 2 retains $(0.8)^t$ of the product's value. Otherwise, the game continues. If no agreement is reached after 10 rounds, the game ends and both players receive 0 units of the resource. We start at $t = 0$. {Player 1/Player 2} makes the first offer. Player 1's utility at time t is $(0.9)^t \times p$ and Player 2's utility at time t is $(0.8)^t \times (100 - p)$ where p is the selling price. Treat this game as infinite horizon for all calculation purposes.
Player 1's goal: Get maximum value out of the sale.
Player 2's goal: Get maximum value out of the sale.

6.2.1 Results

The results of the Rubinstein Bargaining Model for bargaining over a product with utility functions are shown in Table 17. The corresponding metrics for the final agreement are shown in Table 18. The Rubinstein Bargaining solution to this problem is to set the price at 71 when Seller is the first proposer, and 35 when Buyer is the first proposer. The revised prompt significantly enhanced the

Seller	Buyer	Turn 1	Agreement Price	t	Speed
GPT-4	GPT-4	Seller	71	0	33
GPT-4	GPT-4	Buyer	35	0	39
Deepseek R1	Deepseek R1	Seller	71	0	36
Deepseek R1	Deepseek R1	Buyer	35	0	34
GPT-4	Deepseek R1	Seller	71	0	33
GPT-4	Deepseek R1	Buyer	35	0	34
Deepseek R1	GPT-4	Seller	71	0	33
Deepseek R1	GPT-4	Buyer	35	0	35

Table 17: Results of the Rubinstein Bargaining Model for bargaining over a product with utility functions.

Seller	Buyer	Turn 1	Optimality of Seller	Optimality of Buyer
GPT-4	GPT-4	Seller	0	0
GPT-4	GPT-4	Buyer	0	0
Deepseek R1	Deepseek R1	Seller	0	0
Deepseek R1	Deepseek R1	Buyer	0	0
GPT-4	Deepseek R1	Seller	0	0
GPT-4	Deepseek R1	Buyer	0	0
Deepseek R1	GPT-4	Seller	0	0
Deepseek R1	GPT-4	Buyer	0	0

Table 18: Metrics of the Rubinstein Bargaining Model for bargaining over a product with utility functions.

performance of LLMs in the Rubinstein Bargaining Model. The models successfully identified the scenario as a Rubinstein Bargaining Model and consistently reached optimal agreements. By incorporating utility functions into the prompt, the models demonstrated improved reasoning and were able to make accurate inferences. Furthermore, the infinite horizon assumption simplified the analysis, as the models only needed to evaluate the last round, eliminating the need for backward induction from the final round to the first. This adjustment also reduced the computational time required for inference.

7 Conclusion

In this paper, we investigated the bargaining capabilities of LLMs across various game-theoretic scenarios, including the Nash Bargaining Problem, Ultimatum Game, and Rubinstein Bargaining Model. Our evaluation framework analyzed LLMs' performance using metrics such as fairness, rationality, and efficiency, while benchmarking their outcomes against theoretical solutions and existing models. The results indicate that LLMs demonstrate emergent bargaining capabilities, achieving near-optimal agreements in simpler scenarios like the Ultimatum Game. However, they face challenges in more complex setups, such as Rubinstein Bargaining, where they often fail to consistently reach optimal agreements.

Notably, the order of conversation significantly impacted outcomes, with the initiating agent frequently securing a more advantageous deal. While LLMs exhibited strong reasoning in some cases, their limited ability to fully internalize game-theoretic principles highlights gaps in strategic adaptability. Real-world datasets like Amazon History Price further revealed asymmetries in LLMs' performance, excelling as sellers but underperforming as buyers.

We also used prompting methods to improve LLMs' performance in bargaining scenarios. By providing utility functions and examples, we enhanced their understanding of complex bargaining situations, leading to more optimal agreements. The results suggest that LLMs can benefit from structured prompting and reinforcement learning techniques to improve their bargaining strategies.

These findings emphasize the need for targeted enhancements, such as reinforcement learning, adversarial self-play, and AI-feedback mechanisms, to improve LLMs' strategic reasoning. Future research should focus on refining LLMs' ability to navigate dynamic, adversarial, and asymmetric bargaining scenarios, enabling their effective deployment as negotiation agents in real-world applications.

8 Further Work: Improving LLMs' Bargaining Capabilities

8.1 Self-play and Reinforcement Learning

Fu et al. (2023) [10] proposed a reinforcement learning framework, Adversarial Taboo, to enhance LLM reasoning through adversarial self-play. The attacker aims to make the defender unknowingly say a target word, while the defender deduces it. The framework has three stages: (1) imitation learning bootstraps the model using GPT-4 gameplay data, fine-tuning with KL-regularized maximum likelihood to balance strategies and language skills; (2) offline self-play generates diverse adversarial dialogues by alternating attacker and defender roles against a frozen model copy; (3) policy refinement uses Advantage-Leftover-Lunch (A-LoL) for importance-weighted updates and temporal-difference advantage estimation, with Reinforced Self-Training (ReST) filtering high-reward trajectories to stabilize training. This iterative process strengthens reasoning through adversarial competition while maintaining general capabilities.

In bargaining, game-theoretic utility functions serve as proxies for reward functions, encapsulating agent preferences over offers and counter offers. For product bargaining scenarios, the buyer's reward can be modeled as the difference between their maximum willingness-to-pay price and the agreed price, while the seller's reward is represented as the difference between the agreed price and their minimum acceptable price. To discourage overly greedy strategies and promote fairness, the reward function can be refined to measure the difference between the current utility and the optimal utility values derived from theoretical solutions. Additionally, Reinforcement Learning from Human Feedback (RLHF) can be employed to iteratively train LLMs, enabling them to develop more effective and adaptive negotiation strategies over time.

$$reward(s) = \frac{1}{n}\sum_{i=1}^{n}(u_i - u^*)$$

where u_i is the utility of the current state, u^* is the optimal utility, and n is the number of states.

8.2 AI-Feedback and In-Context Learning

To enhance LLMs as negotiation agents, we introduce an AI critic to analyze deliberation dynamics and refine strategies. Abdelnabi et al. [1] suggested using systematic zero-shot Chain-of-Thought (CoT) prompting for LLMs to evaluate multi-agent bargaining scenarios by reasoning through structured examples. Iterative AI feedback loops, where models refine responses based on corrections from more advanced agents, enable improvement over time. Use of supervised learning [21] and in-context learning can help LLMs adapt to dynamic negotiation environments and learn from past interactions.

For our use case, we can leverage the AI critic to systematically evaluate the LLMs' performance in bargaining scenarios. The AI critic can analyze the LLMs' reasoning and decision-making processes by comparing their strategies against theoretical game-theoretic solutions, such as the Nash Bargaining Solution or Rubinstein Bargaining Model. It can identify deviations from optimal strategies, highlight inefficiencies, and provide actionable feedback to refine the LLMs' negotiation approaches.

Additionally, the AI critic can simulate adversarial scenarios by acting as a counterparty with varying levels of rationality or strategic complexity. This allows the LLMs to adapt to diverse negotiation styles and improve their robustness. The feedback loop can be further enhanced by incorporating reinforcement learning techniques, where the AI critic assigns rewards based on the LLMs' adherence to optimal strategies and their ability to achieve favorable outcomes.

By integrating the AI critic into the evaluation framework, we can iteratively improve the LLMs' bargaining capabilities, ensuring they perform effectively across a wide range of scenarios, including those with asymmetric information or dynamic constraints. This approach not only enhances the LLMs' strategic reasoning but also paves the way for their deployment in real-world negotiation applications.

References

[1] Sahar Abdelnabi, Amr Gomaa, Sarath Sivaprasad, Lea Schönherr, and Mario Fritz. LLM-deliberation: Evaluating LLMs with interactive multi-agent negotiation game, 2024.

[2] Gati Aher, Rosa I. Arriaga, and Adam Tauman Kalai. Using large language models to simulate multiple humans and replicate human subject studies, 2023.

[3] Elif Akata, Lion Schulz, Julian Coda-Forno, Seong Joon Oh, Matthias Bethge, and Eric Schulz. Playing repeated games with large language models, 2023.

[4] Valerio Capraro, Roberto Di Paolo, Matjaž Perc, and Veronica Pizziol. Language-based game theory in the age of artificial intelligence. *Journal of The Royal Society Interface*, 21(212), March 2024.

[5] Pengyu Cheng, Tianhao Hu, Han Xu, Zhisong Zhang, Zheng Yuan, Yong Dai, Lei Han, Nan Du, and Xiaolong Li. Self-playing adversarial language game enhances llm reasoning, 2025.

[6] DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025.

[7] Yuan Deng, Vahab Mirrokni, Renato Paes Leme, Hanrui Zhang, and Song Zuo. LLMs at the bargaining table. In *Agentic Markets Workshop at ICML 2024*, 2024.

[8] Haodong Duan, Jueqi Wei, Chonghua Wang, Hongwei Liu, Yixiao Fang, Songyang Zhang, Dahua Lin, and Kai Chen. Botchat: Evaluating llms' capabilities of having multi-turn dialogues, 2023.

[9] Caoyun Fan, Jindou Chen, Yaohui Jin, and Hao He. Can large language models serve as rational players in game theory? a systematic analysis, 2023.

[10] Yao Fu, Hao Peng, Tushar Khot, and Mirella Lapata. Improving language model negotiation with self-play and in-context learning from AI feedback, 2023.

[11] Kanishk Gandhi, Jan-Philipp Fränken, Tobias Gerstenberg, and Noah D. Goodman. Understanding social reasoning in language models with language models, 2023.

[12] Etan A. Green and E. Barry Plunkett. The science of the deal: Optimal bargaining on ebay using deep reinforcement learning. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, EC '22, pages 1–27, New York, NY, USA, 2022. Association for Computing Machinery.

[13] Fulin Guo. Gpt in game theory experiments, 2023.

[14] Mike Lewis, Denis Yarats, Yann N. Dauphin, Devi Parikh, and Dhruv Batra. Deal or no deal? end-to-end learning for negotiation dialogues, 2017.

[15] Zilin Ma, Susannah Cheng Su, Nathan Zhao, Linn Bieske, Blake Bullwinkel, Jinglun Gao, Gekai Liao, Siyao Li, Ziqing Luo, Boxiang Wang, Zihan Wen, Yanrui Yang, Yanyi Zhang, Claude Bruderlein, and Weiwei Pan. Using large language models for humanitarian frontline negotiation: Opportunities and considerations. In *ICML 2024 Next Generation of AI Safety Workshop*, 2024.

[16] Shaoguang Mao, Yuzhe Cai, Yan Xia, Wenshan Wu, Xun Wang, Fengyi Wang, Tao Ge, and Furu Wei. Alympics: Llm agents meet game theory – exploring strategic decision-making with ai agents, 2024.

[17] Y. Narahari. *Game Theory and Mechanism Design*. World Scientific Publishing Company.

[18] John Nash. *2. The Bargaining Problem. Econometrica 18 (1950) 155-162.*, pages 5–13. Princeton University Press, Princeton, 1997.

[19] OpenAI. Gpt-4 technical report, 2024.

[20] Ariel Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50:97–109, 02 1982.

[21] Johannes Schneider, Steffi Haag, and Leona Chandra Kruse. Negotiating with llms: Prompt hacks, skill gaps, and reasoning deficits, 2024.

[22] Richard Thaler. The ultimatum game. *Journal of Economic Perspectives*, 2:195–206, 02 1988.

[23] 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, 2024.

[24] 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, 2024.