

# Mastering Diamonds: Teaching genAI to Play

Bhavika Maini

March 2024

## 1 Introduction

The game of Diamonds presents a strategic challenge where players bid for diamond cards to accumulate points. In this essay, we explore the journey of teaching an AI, termed genAI, to play Diamonds adeptly using optimizing strategies. We'll outline the rules of the game, the prompts provided to genAI, the iterative teaching process, and the strategies employed. Additionally, we'll provide pseudocode for genAI's bidding strategy and analyze its practical results.

## 2 Problem Statement

The objective is to develop an AI capable of playing Diamonds effectively, maximizing its chances of winning through optimal bidding and card-playing strategies. The AI must adapt to various scenarios and opponents, making informed decisions based on available information.

## 3 Rules of the Game

- Each player gets a suit of cards other than the diamond suit.
- The diamond cards are then shuffled and put on auction one by one.
- All the players must bid with one of their own cards face down.
- The banker gives the diamond card to the highest bid, i.e., the bid with the most points.

- Card rankings:  $2 < 3 < 4 < 5 < 6 < 7 < 8 < 9 < T < J < Q < K < A$
- The winning player gets the points of the diamond card to their column in the table.
- If there are multiple players that have the highest bid with the same card, the points from the diamond card are divided equally among them.
- The player with the most points wins at the end of the game.

## 4 Teaching genAI the Game

### 4.1 Understanding the Rules

genAI is introduced to the game's rules, including card rankings, bidding mechanics, and point allocation.

### 4.2 Data Input

genAI is trained on a dataset containing simulated game scenarios, enabling it to learn from diverse situations, bids, and outcomes.

### 4.3 Reinforcement Learning

Through reinforcement learning techniques, genAI learns from the consequences of its actions in simulated games, gradually improving its decision-making abilities.

### 4.4 Prompting for Strategy

genAI is prompted with questions and scenarios to guide its learning process. For instance:

- "What would be your optimal bid given the current bid and your hand?"
- "How would you adjust your strategy if multiple players bid the same card?"
- "What factors influence your decision to bid or pass?"

## 4.5 Iterative Improvement

genAI undergoes multiple training iterations, refining its strategies based on feedback and performance evaluation.

## 5 Strategy Discussed

- **Optimal Bidding:** genAI aims to bid the minimum card necessary to win the diamond, preserving higher-value cards for later rounds.
- **Adaptive Bidding:** It adjusts its bids based on opponents' tendencies and game dynamics, ensuring flexibility in decision-making.
- **Risk Assessment:** genAI evaluates the risk-reward ratio for each bid, considering potential gains versus the cost of using valuable cards.
- **Collaborative Bidding:** In scenarios where multiple players bid the same card, genAI calculates the optimal bid share to maximize its point accumulation while remaining competitive.

## 6 Code for the Strategy

```
# Pseudocode for genAI's bidding strategy
def optimal_bid(current_bid, hand):
    min_bid = min(hand) # Find the minimum card in hand
    if min_bid > current_bid:
        return min_bid # Bid the minimum card if it's higher than the current
    else:
        return None # Pass if the current bid is higher than the minimum card

# Pseudocode for genAI's decision-making strategy when multiple players bid th
def adjust_bid_share(diamond_points, num_players):
    return diamond_points / num_players # Divide diamond points equally among

# Main loop for genAI's gameplay
while game_not_over:
    current_bid = get_current_bid()
    hand = get_hand()
```

```

    if can_bid(current_bid, hand):
        bid = optimal_bid(current_bid, hand)
        if bid is not None:
            place_bid(bid)
    else:
        pass_turn()

# Handle scenarios where multiple players bid the same card
if multiple_players_bid_same_card():
    diamond_points = get_diamond_points()
    num_players = count_bidding_players()
    bid_share = adjust_bid_share(diamond_points, num_players)
    update_points(bid_share)

# End of game: Calculate total points and determine the winner

```

## 7 Practical Results

genAI demonstrates competence in playing Diamonds, effectively bidding and accumulating points based on optimal decisions. Through simulations and actual gameplay, genAI showcases its ability to adapt to varying game conditions and opponents, maximizing its chances of winning.

## 8 Conclusion

Teaching genAI to play Diamonds involved a structured approach of understanding rules, reinforcement learning, and iterative improvement. The implemented strategy focused on optimal bidding, adaptive decision-making, risk assessment, and collaborative bidding. genAI's proficiency in playing Diamonds highlights its potential in strategic game scenarios, paving the way for further advancements in AI gameplay. As AI continues to evolve, continued refinements and optimizations will enhance its performance, offering new opportunities for intelligent gameplay.

## **Thank You**

Thank you for reading!

---