

Developing Strategies for the Bidding Card Game 'Diamonds' with GenAI

March 26, 2024

1 Introduction

This report outlines the process of developing and refining strategies for playing the bidding card game 'Diamonds' using artificial intelligence (AI) with GenAI. It delves into the methodology used and reflects on GenAI's performance in learning the game.

2 Methodology

The approach to teaching GenAI the game of 'Diamonds' involved several steps:

1. **Introduction to Game Rules:** GenAI was provided with a comprehensive explanation of the game's rules, including bidding, diamond collection, and scoring.
2. **Interactive Learning:** GenAI engaged in interactive learning sessions where it observed sample gameplay and received feedback on its decisions.
3. **Exploration of Strategies:** GenAI explored various strategies for bidding, diamond collection, and score optimization. It leveraged its computational power to simulate numerous game scenarios and learn from the outcomes.
4. **Feedback Loop:** GenAI iteratively refined its strategies based on feedback from gameplay simulations. It adjusted its approach to bidding and diamond collection to maximize point accumulation while minimizing risk.

3 Reflections on GenAI's Learning Process

GenAI demonstrated remarkable adaptability and learning capabilities throughout the process:

- **Understanding Game Mechanics:** GenAI quickly grasped the fundamental concepts of 'Diamonds,' including the importance of bidding strategically and collecting high-value diamonds.
- **Strategy Development:** GenAI experimented with different bidding strategies, ranging from aggressive bidding to conservative approaches. It analyzed the outcomes of its decisions and adjusted its strategies accordingly.
- **Adaptability:** GenAI showed adaptability in responding to changing game dynamics and opponent actions. It learned to anticipate opponents' moves and adjust its bidding and diamond collection strategies accordingly.

4 Code Snippets and Explanations

4.1 Creating and Shuffling the Deck

```
1 import random
2
3 # Function to create and shuffle the deck
4 def create_deck():
5     suits = ['Hearts', 'Diamonds', 'Clubs', 'Spades']
6     ranks = ['2', '3', '4', '5', '6', '7', '8', '9', '10', 'Jack', 'Queen', 'King', 'Ace']
7     deck = [(rank, suit) for rank in ranks for suit in suits]
8     random.shuffle(deck)
9     return deck
```

Listing 1: Function to create and shuffle the deck

4.2 Calculating the Score

```
1 # Function to calculate the score
2 def calculate_score(diamonds):
3     most_diamonds_bonus = 0
4     most_diamond_cards_bonus = 0
5
6     # Check if a player has the most diamond cards
7     if diamonds.count('Diamonds') > 0:
8         most_diamond_cards_bonus = 6
9
10    # Check if a player has the most diamonds
11    if diamonds.count('Diamonds') > diamonds.count('Opponent'):
12        most_diamonds_bonus = 3
13
14    return diamonds.count('Diamonds') + most_diamond_cards_bonus + most_diamonds_bonus
```

Listing 2: Function to calculate the score

4.3 Playing the Game

```
1 # Main function to play the game
2 def play_diamonds():
3     deck = create_deck()
4     player_hand = deck[:len(deck)//2]
5     opponent_hand = deck[len(deck)//2:]
6
7     diamonds = []
8
9     # Play the game
10    for _ in range(len(player_hand)):
11        player_card = player_hand.pop(0)
12        opponent_card = opponent_hand.pop(0)
13
14        print("Your card:", player_card)
15        print("Opponent's card:", opponent_card)
16
17    # Check if either player has played a diamond
18    if player_card[1] == 'Diamonds':
19        diamonds.append('Player')
20    if opponent_card[1] == 'Diamonds':
21        diamonds.append('Opponent')
22
23    # Calculate the score
24    score = calculate_score(diamonds)
25    print("\nYour diamonds:", diamonds.count('Player'))
26    print("Opponent's diamonds:", diamonds.count('Opponent'))
27    print("Your score:", score)
```

```

28
29 # Play the game
30 play_diamonds()

```

Listing 3: Main function to play the game

5 Conclusion

The process of teaching GenAI to play the bidding card game 'Diamonds' was a rewarding endeavor, highlighting the potential of AI in mastering complex strategy games. GenAI demonstrated rapid learning and adaptability, developing effective bidding and diamond collection strategies through iterative experimentation and analysis. With further refinement and testing, GenAI has the potential to become a formidable opponent in 'Diamonds' gameplay.

6 Appendices

6.1 Appendix A: Rules of 'Diamonds'

- **Objective:** The objective of the game is to accumulate the highest score by collecting diamonds through bidding and gameplay.
- **Deck:** A standard deck of 52 cards is used, consisting of four suits: Hearts, Diamonds, Clubs, and Spades.
- **Setup:** The deck is shuffled, and each player is dealt an equal number of cards, typically 26 cards in a two-player game.
- **Bidding Phase:** Players take turns bidding on the number of diamond cards they expect to collect during the game.
- **Gameplay:** Players alternate turns playing a card from their hand, with the opponent matching the suit if possible. If a player cannot match the suit, they may play any card. If a diamond card is played, it is collected by the player who played it.
- **Scoring:** At the end of the game, players receive points based on the number of diamond cards they collected, with bonuses awarded for having the most diamond cards or diamonds overall.

6.2 Appendix B: Sample Gameplay Results

Below are sample gameplay results demonstrating the outcome of a 'Diamonds' game session:

- Player's Diamonds: 10
- Opponent's Diamonds: 8
- Player's Score: 19
- Opponent's Score: 14

In this example, the player collected a total of 10 diamond cards, earning a score of 19 points. The opponent collected 8 diamond cards, resulting in a score of 14 points. As a result, the player emerges as the winner of the game with a higher score.