Knowledge-Based Systems Final Report

GTO Poker Bot: A Knowledge-Based Approach to Optimal Poker Strategy

Arne Noori, Ido Pesok & Wes Convery

1. Introduction

This report presents GTO Poker Bot, an AI system designed to find an optimal poker strategy using a combination of knowledge-based approaches and fixed strategies. The bot operates at the browser level, simulating human-like gameplay by interpreting the game state visually. It offers two agent types: a fixed strategy agent based on game theory optimal principles and a Deep Q-Network (DQN) agent that learns through self-play. The project aims to explore the potential for AI to solve the complex game of poker, which, unlike chess and Go, remains an unsolved challenge. The bot is implemented in Python and can be easily adapted to work on various poker platforms. The focus of this report is on the knowledge-based aspects of the project, particularly the fixed strategy agent and its performance in comparison to the DQN agent.

Poker presents a unique challenge for artificial intelligence due to its elements of imperfect information, stochasticity, and game-theoretic complexity. While AI has achieved superhuman performance in perfect information games like chess (BigFish) and Go (AlphaZero), poker remains an unsolved problem with room for further strategic optimization. This project, GTO Poker Bot, aims to tackle this challenge by developing an AI system that can learn and execute optimal poker strategies through a combination of knowledge-based approaches and reinforcement learning.

The primary motivation behind this project is to explore the potential for AI to master the intricate decision-making processes required in poker. By creating a bot that operates at the browser level and interprets the game state visually, we simulate human-like gameplay, enabling the AI to interact with various poker platforms without relying on direct access to the game's internal state. This approach opens up possibilities for testing and refining poker AI in real-world settings.

The GTO Poker Bot offers two distinct agent types: a fixed strategy agent and a DQN agent. The fixed strategy agent is the focus of this report, as it embodies the knowledge-based aspects of the project. This agent bases its decisions on game theory optimal principles, executing a predefined strategy that takes into account hand strength, pot odds, and opponent actions. By comparing the performance of this agent to the DQN agent, we can evaluate the effectiveness of a knowledge-based approach in approximating optimal poker strategy.

The implementation of GTO Poker Bot utilizes Python and can be easily adapted to work on various poker platforms. Users can run the "play.py" script to observe the bot playing poker autonomously, making decisions based on the visual interpretation of the game state.

2. Problem Specification

Problem Description

The project aims to address the challenge of creating an intelligent agent capable of playing poker autonomously by simulating human-like interactions with a web browser. The goal is to navigate the complexities of poker, a game characterized by incomplete information, strategic depth, and the necessity for bluffing and adaptation. Unlike Chess or Go, which have seen AI achieve superhuman performance, poker remains a largely unsolved domain, offering a fertile ground for AI research and development.

Main Challenges

1. **Incomplete Information:** Unlike deterministic games with perfect information, poker involves hidden information (opponent's cards), making it challenging to predict outcomes accurately.
2. **Dynamic Strategy:** The necessity to adapt strategies based on the evolving game state and opponents' actions.
3. **Simulating Human Interaction:** Developing an agent that can interact with web-based poker platforms as a human would, through vision and action.

Progress in the Domain

AI research in poker has made significant strides, with notable projects like DeepStack and Libratus demonstrating the potential for AI to compete at high levels. These projects have utilized deep learning and game theory to navigate the complexities of poker, providing a foundation for further exploration.

Tools and Starting Points

The project leverages TensorFlow for implementing the DQN agent and GPT-V for interpreting the game state from screenshots. The fixed strategy agent [fixed.py] and the DQN agent [dqn_agent.py] serve as the core components, with the operational framework [play.py]) enabling interaction with online poker games.

PEAS Specification (Simplified)

- **Performance:** Success is measured by the agent's ability to win hands and maximize earnings over time.
- **Environment:** Online poker platforms, characterized by varying game states, opponent strategies, and the visual presentation of information.
- **Actuators:** Actions within the game, such as fold, call, and raise, executed through simulated mouse clicks and keyboard inputs.
- **Sensors:** Screenshots of the game, processed and interpreted to understand the current state of the board and make decisions.

Game Rules and Parameters

Poker, specifically the Texas Hold'em variant, involves players making the best hand from two private cards and five community cards. The game consists of several betting rounds (preflop, flop, turn, river), with players having the option to fold, call, or raise. The complexity of poker comes from the strategic decisions made with incomplete information about opponents' hands.

Implementation-Specific Parameters

- **Number of Cards:** 52-card deck.
- **Starting Stack:** Each player begins with a predetermined amount of chips.
- **Blinds:** Small and big blinds are posted at the beginning of each hand to initiate betting.

3. Related Work

Several notable poker AI systems have been developed in recent years, each employing different approaches to tackle the complexities of the game. Libratus, developed at Carnegie Mellon University, uses a form of counterfactual regret minimization (CFR+) to compute strategies in real-time [1]. Pluribus, created by researchers from Facebook AI and Carnegie Mellon, builds upon the success of Libratus to beat elite human pros in multiplayer no-limit Texas hold'em [2]. DeepStack, developed by researchers from the University of Alberta and Czech Technical University, uses deep learning and decomposition to reason about game situations [13].

While these state-of-the-art systems have achieved remarkable success, they rely on sophisticated techniques that may be challenging to implement and computationally expensive. In contrast, the GTO Poker Bot explores the potential of a knowledge-based approach, particularly through the use of a fixed strategy agent. This agent embodies the principles of game theory optimal play, providing a strong baseline for evaluating the performance of the learning agent (DQN).

The fixed strategy agent draws inspiration from the concept of Nash equilibrium, a central concept in game theory. In a two-player zero-sum game like heads-up poker, a Nash equilibrium is a pair of strategies where neither player can improve their expected outcome by unilaterally changing their strategy [14]. The fixed strategy agent aims to approximate a Nash equilibrium strategy by making decisions based on hand strength, pot odds, and opponent actions.

The implementation of the fixed strategy agent builds upon existing knowledge in the field of poker game theory. Resources such as GTO Wizard [15] and GTOBase [16] provide insights into game theory optimal strategies for various poker situations. By incorporating this knowledge into the fixed strategy agent, the GTO Poker Bot explores the effectiveness of a knowledge-based approach in achieving strong poker performance.

4. Implementation

Methods and Justification

1. **Fixed Strategy Agent (`fixed.py`):** This agent follows a predetermined set of rules derived from GTO principles. It evaluates the current state of the game, including the community cards, hole cards, pot value, and opponents' actions, to make decisions that align with a fixed strategy deemed optimal based on game theory.
2. **DQN Agent (`dqn_agent.py`):** The DQN agent utilizes a reinforcement learning model to learn from each game's outcomes. It dynamically adjusts its strategy based on the rewards or penalties received from previous actions, aiming to maximize its winnings over time. This agent represents a more flexible and adaptive approach, capable of learning from experience and refining its strategy accordingly.

The operational framework, as demonstrated in [play.py], integrates these agents with a system that captures and interprets the game's state from screenshots using GPT-V. This innovative approach allows the agent to "see" the game similarly to a human player, making decisions based on the visual information available. The framework then translates these decisions into actions executed within the browser, enabling the agent to play poker autonomously on virtually any online platform.

Implementation

- **Resources**: Our implementation builds upon existing research and tools in the field of AI poker. We referenced academic papers and utilized open-source libraries for reinforcement learning and game simulation. Specific references include works by Brown and Sandholm on superhuman AI for poker, and tools like GTO Wizard and GTOBase for understanding GTO strategies.
- **Knowledge Representation**: Our system handles knowledge of poker rules, strategies, and game states. This knowledge is obtained through a combination of hardcoded rules (for the fixed strategy agent) and data collected from simulated games (for the DQN agent). The knowledge is represented using custom classes and objects in Python, with game states stored in JSON format for ease of manipulation and analysis.
- **System Description**: The system is composed of two main modules: the agents (fixed strategy and DQN) and the operational framework that enables interaction with online poker platforms. The DQN agent's training procedure involves playing thousands of simulated games, with the model learning from the outcomes of these games. The fixed strategy agent does not require training.
- **Evaluation**: The system's performance is evaluated by simulating games against both random agents and agents following fixed strategies. For the DQN agent, we track the improvement in win rate over time as an indicator of learning. The fixed strategy agent's performance is evaluated based on its consistency and ability to achieve a positive win rate against a predefined set of opponents.

- **Training Data for DQN Agent**: The DQN agent was trained on a dataset of 1000 simulated poker hands. The loss function used was the mean squared error between the predicted and actual outcomes, with the aim of minimizing this error over time.

This implementation showcases the potential of AI in mastering complex games like poker, highlighting both the strengths and limitations of fixed strategies and adaptive learning approaches.

5. Analysis

Fixed Strategy Agent (fixed.py): The fixed strategy agent is the embodiment of the knowledge-based approach in the GTO Poker Bot. This agent makes decisions based on a predefined set of rules derived from game theory optimal principles. The agent considers factors such as hand strength, pot odds, and opponent actions to determine the optimal action in a given situation.

The implementation of the fixed strategy agent involves the following key components: 1. Hand Strength Evaluation: The agent assesses the strength of its hand based on the community cards and its hole cards. It uses a heuristic scoring system that assigns higher values to stronger hands (e.g., royal flush, straight flush, four of a kind) and lower values to weaker hands (e.g., high card, one pair).

2. Pot Odds Calculation: The agent calculates the pot odds, which represent the ratio of the current size of the pot to the cost of a contemplated call. This information is used to determine whether it is profitable to continue in the hand or fold.

3. Opponent Action Consideration: The agent takes into account the actions of the opponents, such as their betting patterns and the frequency of their raises. This information is used to adjust the agent's strategy and make more informed decisions.

4. Decision Making: Based on the hand strength, pot odds, and opponent actions, the agent selects the optimal action from a predefined set of options (e.g., fold, call, raise). The decision-making process is guided by a set of rules that aim to maximize the expected value of the agent's actions.

The fixed strategy agent serves as a strong baseline for evaluating the performance of the DQN agent. By comparing the results of the two agents, we can assess the effectiveness of the knowledge-based approach and identify areas for improvement.

Evaluation:

To evaluate the performance of the GTO Poker Bot, particularly the fixed strategy agent, we conducted a series of simulations and analyses.

1. Head-to-Head Comparison: We pitted the fixed strategy agent against the DQN agent in a series of head-to-head matches. The agents played a total of 1000 hands, and their performance was measured in terms of the number of hands won, the total chips accumulated, and the overall win rate.

2. Performance Against Human Players: To assess the effectiveness of the fixed strategy agent in real-world scenarios, we conducted a series of matches against human players of varying skill levels. The agent's performance was evaluated based on its ability to make profitable decisions and adapt to different playing styles.

3. Comparison to Game Theory Optimal Strategies: We compared the decisions made by the fixed strategy agent to the game theory optimal strategies derived from resources such as GTO Wizard [15] and GTOBase [16]. This analysis allowed us to evaluate the extent to which the agent's knowledge-based approach approximates optimal play.

4. Robustness and Adaptability: We tested the fixed strategy agent's robustness by exposing it to a wide range of game situations and opponent strategies. This evaluation helped identify the strengths and limitations of the knowledge-based approach and highlighted areas for improvement.

The evaluation results demonstrated that the fixed strategy agent achieved a strong performance against both the DQN agent and human players. The agent's knowledge-based approach, based on hand strength, pot odds, and opponent actions, proved effective in making profitable decisions and adapting to different game situations.

However, the evaluation also revealed some limitations of the fixed strategy approach. The agent's performance was somewhat constrained by its reliance on predefined rules and heuristics. In certain complex game situations, the agent struggled to make optimal decisions, highlighting the need for more advanced techniques such as real-time adaptation and opponent modeling. for AI to master the strategic and uncertain environment of online poker.

6. Ethical Considerations

The development and deployment of an autonomous poker-playing bot, particularly one capable of bypassing cheat detection mechanisms on websites, raises several ethical considerations. These considerations are not only relevant to the realm of online poker but extend to the broader context of AI applications in various online platforms. Here are some of the key ethical concerns:

Fairness and Integrity in Gaming

- **Impact on Fairness**: The use of an autonomous bot in online poker disrupts the level playing field expected by human players. Poker, like many games, is predicated on human skill, psychology, and unpredictability. Introducing an AI that can play autonomously—and potentially more effectively than most humans—undermines the spirit of fair competition.
- **Violation of Terms of Service**: Most online gaming platforms explicitly prohibit the use of automated systems or bots. Using such technology not only breaches these terms but also places the user at risk of penalties, including account suspension or legal action. **Transparency and Disclosure:** If the GTO Poker Bot is deployed in real-world poker environments, it is important to disclose its presence to the other players. Transparency about the use of AI agents promotes trust and allows players to make informed decisions about their participation.

Security and Privacy Concerns

- **Bypassing Cheat Detection**: The ability of the bot to evade detection mechanisms can be seen as a direct challenge to the security measures implemented by online platforms. This capability might encourage the development and use of similar technologies in other contexts, potentially leading to widespread exploitation.
- **Data Privacy**: The operation of such bots may involve analyzing game data, player behavior, and potentially sensitive information. Ensuring the privacy and security of this data is paramount to prevent misuse.

Societal Impact

- **Normalization of Cheating**: The existence and use of such bots could contribute to a broader normalization of cheating and unethical behavior in online environments. This could erode trust in online platforms and digital interactions more generally.
- **Economic Impact**: For many, online poker is not just a game but a source of income. The widespread use of poker bots could disrupt the economy of these games, affecting the livelihoods of professional players and the revenue of platforms that host these games.

Mitigation Strategies

To address these ethical considerations, several steps can be taken:

- **Transparency and Consent**: If used for research purposes, it's crucial to operate in environments where all participants are aware of and consent to the involvement of AI agents.
- **Compliance with Legal and Ethical Standards**: Developers should ensure their projects comply with the legal and ethical standards set by both the platforms and the broader community. This includes respecting terms of service and privacy policies.

- **Responsible Disclosure**: If the technology developed has the potential to exploit vulnerabilities in online platforms, it's ethical to disclose these findings responsibly to the affected parties to allow them to strengthen their security measures.
- **Public Discussion and Regulation**: Engaging in public discourse about the implications of such technologies can help shape guidelines and regulations that balance innovation with ethical considerations.

In conclusion, while the development of an autonomous poker-playing bot using advanced AI techniques like GPT-4 vision calls represents a significant technological achievement, it also underscores the need for a careful, ethical approach to AI development and deployment. As AI continues to evolve, so too must our ethical frameworks and policies to ensure these technologies benefit society without compromising fairness, privacy, or security.

7. Link to code

https://github.com/arnenoori/gto-poker-bot

Tutorial on how to run the code and replicate our results are within the README.md file on the repository.

8. Summary

Conclusion: The GTO Poker Bot project demonstrates the application of knowledge-based approaches and fixed strategies in the pursuit of optimal poker play. By combining game theory principles with visual interpretation and autonomous decision-making, the bot offers a novel approach to tackling the complexities of poker.

The fixed strategy agent, which embodies the knowledge-based aspects of the project, achieved strong performance against both the DQN agent and human players. Its decision-making process, based on hand strength, pot odds, and opponent actions, proved effective in making profitable decisions and adapting to different game situations.

However, the evaluation also highlighted the limitations of the fixed strategy approach, particularly in handling complex game situations and adapting to diverse opponent strategies. These limitations underscore the need for more advanced techniques, such as real-time adaptation and opponent modeling, to further enhance the bot's performance.

The project also raises important ethical considerations, such as fairness, transparency, and responsible use. Addressing these ethical aspects is crucial to ensure the integrity of the online poker ecosystem and maintain trust among players.

Moving forward, the GTO Poker Bot project offers several avenues for further research and development. Integrating more advanced techniques, such as opponent modeling and real-time adaptation, could help overcome the limitations of the fixed strategy approach. Exploring the combination of knowledge-based approaches with reinforcement learning could also lead to more robust and adaptive poker AI systems.

In conclusion, the GTO Poker Bot project demonstrates the potential of knowledge-based approaches and fixed strategies in developing effective poker AI systems. While challenges remain, the project lays the foundation for further research and development in the field of poker AI, contributing to the ongoing quest for optimal poker play.

9. References

[1] Brown, Noam, and Tuomas Sandholm. "Superhuman AI for Heads-up No-Limit Poker: Libratus Beats Top Professionals." Science, 17 Dec. 2017.

[2] "Pluribus: The First AI to Beat Pros in 6-Player Poker." Meta AI, 11 July 2019, ai.meta.com/blog/pluribus-first-ai-to-beat-pros-in-6-player-poker.

[3] Moravčík, Matej, et al. "DeepStack: Expert-Level Artificial Intelligence in Heads-up No-Limit Poker." Science, vol. 356, no. 6337, 2017, pp. 508-513.

[4] Nash, J. F. "Equilibrium Points in N-Person Games." Proceedings of the National Academy of Sciences, vol. 36, no. 1, 1950, pp. 48-49.

[5] "GTO Wizard: GTO Trainer for Cash Games & Tournaments." GTO Wizard, www.gtowizard.com.

[6] "GTO Poker Strategy Viewer, Trainer and HH Analyzer." GTOBase, 20 Jan. 2023.