Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/33/20230256

Research of artificial intelligence in imperfect information card games

Megan Sun

Westlake High School, Thousand Oaks, 91362, USA

126652@learn.conejousd.net

Abstract. Artificial intelligence (AI) in games has advanced significantly, notably in perfect information games such as Go and Chess. Imperfect information games, in which participants do not have complete information about the game state, create more difficulties. They incorporate both public and private observations, where strategies must be improved to achieve a Nash equilibrium. This study investigates artificial intelligence and reinforcement learning approaches, in which agents learn to maximize future rewards through interactions with their surroundings. The paper then focuses on card game research platforms such as RLCard and OpenAI Gym. It gives a comprehensive summary of research in No Limit Texas Hold'em, a difficult two-player poker game with a large decision space. DeepStack and Libratus are successful systems that have attained expert-level and superhuman play, respectively. Pluribus, a superhuman artificial intelligence for six-player poker, and DouZero, a pure reinforcement learning technique for the multiplayer card game, DouDiZhu, are both investigated. Overall, this paper provides background information on reinforcement learning and imperfect information games, analyzes commonly used research platforms, evaluates the effectiveness of AI algorithms in various card games, and offers future research areas and directions.

Keywords: artificial intelligence, reinforcement learning, counterfactual regret minimization, card games.

1. Introduction

Artificial intelligence (AI) in games is a widely researched topic, as it provides an environment to improve and enhance machine learning algorithms. Computers have reached superhuman levels of play in perfect information games, where each player is informed of the complete state of the game and the history of events. Examples include IBM's Deep Blue [1] in chess, Google's Alpha Go [2] and Alpha Zero [3] in Go, and IBM's Watson in Jeopardy [4]. Conversely, card games are often imperfect information games. Certain information can be hidden, including the opponent's hand, past cards played, and the order of cards left in the deck. The property of hidden information requires more complex reasoning compared to similar-sized perfect information games. The correct action for an agent to take at a certain state depends on the probability of what cards the opponent has, which is reliant on their previous actions. Although, the action that the opponent chooses is based on their knowledge of our private cards, which is dependent on our past actions. This causes recursive-based belief reasoning, which makes it impossible to think about situations in isolation. AI either must use approaches that reason about the complete game before beginning, such as counterfactual regret minimization (CFR),

DOI: 10.54254/2755-2721/33/20230256

or develop modifications and new techniques to decrease computational cost [5]. In addition, card games can have in extremely large state spaces, suffer from sparse rewards, and be played by several agents who cooperate or compete with one other. These unique properties make research in card games and imperfect information games particularly applicable to the real world, as they may be viewed as abstractions of many current problems.

2. Reinforcement learning

Reinforcement Learning is the art of decision-making. The goal of the agent is to choose an action that maximizes its future rewards [6]. It is not explicitly told which actions to take in certain states, but instead learns an optimal policy through interactions with its environment. Problems are usually represented mathematically as Markov Decision Processes. The cumulative reward is defined as $G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} \ldots = \Sigma_{i=0}^{T} \gamma^i R_{t+i+1}$ (γ is the discount factor) and the state-action value function under policy π is defined as $q_\pi(s,a) = E_\pi[G_t \mid S_t = s, A_t = a]$.

3. Imperfect information games

Imperfect information games contain both public and private observations. For example, in poker, public observations include the bets placed by each player and private observations include what cards they have in their hand. Each player has access to its own information states and chooses an optimal policy π that maximizes its future rewards. A strategy profile contains every player's strategy, defined as $\pi = (\pi_1,\ldots,\pi_n)$. A Nash equilibrium is achieved when nothing can be gained if a player deviates from their strategy. Therefore, it indicates that each strategy is the best response to every other strategy in the strategy profile. Exploitability is defined as the deviation from the Nash equilibrium [7]. If the state space is too large and the computer is not able to calculate the NE strategy, it is optimal to minimize the exploitability.

4. Card game research platforms

RLCard provides an interface to research reinforcement learning algorithms in card games such as Dou Dizhu, Texas Hold'em, Blackjack, Leduc Hold'em, and Mahjong [8]. Algorithms including Deep Q-Network, Counterfactual Regret Minimization, and Neural Fictitious Self-Play have been effectively used in the environment.

A more general research platform for imperfect information games is OpenAI Gym. Its initial release contained a toolkit that provided five environments: Atari, Algorithmic, Board Games, Classic control and toy text, and 2D and 3D robots [9]. However, it has been extended to many multi-agent settings and card games since then.

5. Case study: no limit texas hold'em

Each player in a game of No Limit Texas Hold'em initially receives two cards face down, with the option to check, bet, or fold. Three rounds of betting follow, and additional cards are revealed. With no limit on the betting amount, the hand ends when every player but one folds or cards are turned over and the best hand wins.

The two-player version of the game has around 10160 decision points [10], exceeding the limits of best approximate equilibrium-finding algorithms. For instance, counterfactual regret minimization has a limit of 1012 decision points [5]. However, in heads-up limit Texas Hold'em (HUNL), a simpler variation with fixed bets and just under 1014 decision points, computers have found success using counterfactual regret minimization plus (CFR+) [11]. To solve no-limit Texas Hold'em, computerized abstraction techniques have been researched to reduce the state space and number of decision points. However, abstraction results in a loss of information that can hinder the effectiveness of the algorithm and requires expert-level knowledge which is often costly to obtain.

DeepStack, introduced by Moravčík et al reached expert-level play using CFR to account for information asymmetry, without implementing abstraction techniques [12]. It avoids calculating its complete strategy before the game to achieve a low-exploitability strategy. Instead, it uses continual

DOI: 10.54254/2755-2721/33/20230256

resolving to reconstruct the solution strategy as each public state arises. Though sound in theory, continual resolving only works in practice when the game is almost over. In order to reduce computation, DeepStack utilizes a learned counterfactual value function to replace subtrees at a certain depth. It also limits the set of actions considered using sparse look-ahead trees. DeepStack is trained on ten million random poker hands and uses a conventional feedforward neural network (seven layers with 500 nodes per layer). The pot size and each player's ranges serve as the inputs, while the outputs are the counterfactual values for each player (represented as fractions of the pot size). In 3000 games against professional poker players, 10 of the 11 players were beaten with statistical significance.

Concurrently, Brown and Sanholm developed Libratus, which reached superhuman levels of play [13]. It is split into three modules. The first module creates a blueprint strategy based on an abstraction of the game. This strategy is temporarily used in earlier rounds, computed using a more advanced form of MCCR (Monte Carlo Counterfactual Regret Minimization) [14]. As the game progresses, the second module solves a finer-grained abstraction of the subgame. If the opponent's action is not present in the abstraction, it uses nested subgame solving to include it in the solution. However, a subgame can't be independently solved since it needs to factor in other unreachable subgames to avoid exploitability. Therefore, Libratus incorporates the new strategy into the overarching blueprint strategy. The third module, the self-improvement module, looks for the most common off-blueprint actions and fills in missing information in the blueprint accordingly. This algorithm avoids the problem of lost information due to abstraction while also maintaining computational practicability. Similar to DeepStack's continual resolving, Libratus uses nested subgame solving to compute strategies in real time. However, Libratus uses its blueprint strategy in earlier rounds of the game, compared to DeepStack's depth-limited approach. Libratus defeated a team of 4 HUNL specialists with 99.98% statistical significance.

In 2019, Brown and Sanholm released Pluribus, a superhuman AI for six-player poker [15]. Similar to Libratus, Pluribus initially creates a blueprint strategy to play the first betting round. However, due to the increased number of players and amount of hidden information, it isn't feasible to solve to the end of subgames in the same way as Libratus. Instead, it changes strategies based on which part of the game the agent is in and the size of the subgame. If it is relatively early in the game or the size of the subgame is relatively small, it uses Monte Carlo Linear CFR. Otherwise, a vector-based variation of Linear CFR that has been optimized is used. Pluribus was evaluated against poker professionals in 10,000 hands, using AIVAT to reduce the luck factor [16]. Pluribus was able to win with 95% statistical significance.

6. Case study: doudizhu (fighting the landlord)

The game is played with one deck of 54 cards. 17 cards are given to each three players and 3 remaining cards are left face down. An auction takes place to determine who will become the landlord and pick up the 3 extra cards. The other 2 will team up as peasants. The landlord gets to play first and choose from various moves including a single, pair, triple, sequence, sequence of pairs, bomb, rocket, and quads. Each subsequent player plays a combination of the same type or passes. When two consecutive players pass, the round ends. The player that runs out of cards first wins.

This game is especially challenging because of two factors. The first is the cooperation between the peasants. On certain occasions, one peasant must support the other, playing a card that is not necessarily good for themselves as an individual. In addition, the sizable number of card combinations may lead to a large action space. It is difficult for any sort of abstraction since even taking away one card can lead directly to a loss. This greatly increases the computational cost.

Approaches using Deep Q Networks and A3C have not been effective regarding the game of DouDizhu. It was shown by You et al., 2019 that this algorithm only had a 30% winning rate against humans when the computer played as peasants [17]. DeltaDou achieved expert-level play by incorporating Fictitious Play MCTS and a policy-based inference algorithm into a framework similar to AlphaZero [18]. However, the methods used by DeltaDou are computationally costly and rely on human abstractions.

DouZero, developed by Zha et al., 2021, was able to beat previous AI programs using a pure reinforcement learning method [19]. It employs classic Monte Carlo methods in addition to encoding,

DOI: 10.54254/2755-2721/33/20230256

parallel actors, and deep neural networks. Monte Carlo methods were suitable because DouDiZhu is an episodic game that can be easily parallelized. DouZero represents the cards by encoding each card combination and action with a 4×15 matrix. The neural architecture uses long short-term memory (LSTM) to encode historical moves. The multi-layer perceptron is then used to generate the Q-values. In order to estimate the target values, the learner updates three global Q-networks, one for each of the three players, using Mean Squared Error (MSE) loss. Three local Q-networks that are routinely synchronized with the global networks are also maintained by each actor. The actors frequently sample game engine trajectories, compute cumulative rewards for each state-action combination, and communicate with the learner via three common buffers. These buffers are separated into numerous entries, each of which contains several data instances. DouZero is model-free and doesn't incorporate search. It only needed 3 days to outperform DeltaDou, which is initialized with heuristics and requires 2 months of training.

7. Conclusion

This paper examines artificial intelligence in imperfect information card games, revealing the problems and accomplishments in the field. Commonly used platforms for card game research include RLCard and Open AI Gym. Furthermore, the case studies of Texas Hold'em and DouDiZhu are investigated in depth. Texas Hold'em previously relied on abstraction due to the lack of computational power and large number of decision points. DeepStack achieved expert-level play in the absence of human abstraction. It created a low exploitable strategy using CFR, continual resolving, and a learned counterfactual value function to replace subtrees at a certain depth. Libratus reached superhuman levels of play using MCCR, an initial blueprint strategy, nested subgame solving, and a self-improvement module. Pluribus was a superhuman AI developed for 6-player poker as an improved version of Libratus. It employed similar algorithms but required much less computing power and training time. Similar advancements have been made in the card game, DouDiZhu. The game is especially challenging to solve because of its team dynamics and large action space. Deep Q-Networks and A3C have proved to be ineffective. Although, DouZero was able to achieve a significantly higher winning rate by improving on previous Monte-Carlo methods, adding action encoding, deep neural networks, and parallel actors.

However, there are still limitations, especially in games with enormous decision spaces and sophisticated cooperative dynamics. The possibility of lost knowledge due to abstraction, as well as the computational expense of efficient techniques, necessitates more investigation. The use of artificial intelligence in incomplete information card games demonstrates AI's potential and its influence on both classical games and wider problem-solving settings. As technology advances, more breakthroughs are anticipated in this research area.

References

[1] Campbell, M., Hoane Jr, A. J., & Hsu, F. H. 2002 *Artificial intelligence* **134(1-2)** 57-83
[2] Silver, D., Huang, A., Maddison, C., et al. 2016 *Nature* **529** 484–489
[3] Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., & Hassabis, D. 2017 a*rXiv*
[4] Ferrucci, David & Levas, Anthony & Bagchi, Sugato & Gondek, David & Mueller, Erik 2013 *Artificial Intelligence* **199-200** 93-105
[5] Zinkevich, M., Johanson, M., Bowling, M., & Piccione, C. 2007 *Advances in neural information processing systems 20*
[6] Kaelbling, L. P., Littman, M. L., & Moore, A. W. 1996 *Journal of artificial intelligence research* **4** 237-285
[7] T. Davis, N. Burch, and M. Bowling 2014 *Proceedings of the Twenty-Eighth Conference on Artificial Intelligence (AAAI)* 630–636
[8] Zha, D., Lai, K. H., Cao, Y., Huang, S., Wei, R., Guo, J., & Hu, X. 2019 *arXiv*
[9] Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. 2016 *arXiv*

DOI: 10.54254/2755-2721/33/20230256

[10] Johanson, M. 2013 *arXiv*
[11] Bowling, M., Burch, N., Johanson, M., & Tammelin, O. 2015 *Science* **347(6218)** 145-149
[12] Matej Moravčík et al. 2017 *Science* **356** 508-513
[13] Noam Brown Tuomas Sandholm 2018 *Science* **359** 418-424
[14] Gibson, R., Lanctot, M., Burch, N., Szafron, D., & Bowling, M. 2012 *Proceedings of the AAAI Conference on Artificial Intelligence* **26(1)** 1355-1361
[15] Noam Brown Tuomas Sandholm 2019 *Science* **365** 885-890
[16] Burch, N., Schmid, M., Moravcik, M., Morill, D., & Bowling, M. 2018 *Proceedings of the AAAI Conference on Artificial Intelligence* **32(1)**
[17] You, Y., Li, L., Guo, B., Wang, W., and Lu 2019 *arXiv*
[18] Jiang, Q., Li, K., Du, B., Chen, H., & Fang, H. 2019 *IJCAI* 1265-1271
[19] Zha, D., Xie, J., Ma, W., Zhang, S., Lian, X., Hu, X., & Liu, J. 2021 a*rXiv*