SARTRE: System Overview

A Case-Based Agent for Two-Player Texas Hold'em

Jonathan Rubin and Ian Watson

Department of Computer Science
University of Auckland, New Zealand
jrub001@aucklanduni.ac.nz, ian@cs.auckland.ac.nz

Abstract. SARTRE (Similarity Assessment Reasoning for Texas hold'em via Recall of Experience) is a `heads-up` (two-player) poker-bot that plays `limit` Texas Hold'em using the case-based reasoning methodology. This paper presents an overview of the SARTRE system. As far as we are aware SARTRE is the only poker-bot designed specifically to play `heads-up` Texas Hold'em using a CBR foundation. The design and implementation of the current system is discussed. Case features are illustrated and their reasons for selection are addressed. Finally, avenues for future areas of investigation are then listed.

1 Introduction

This paper will describe the design and implementation of a heads-up (two-player) poker-bot that plays limit Texas Hold'em using the case-based reasoning methodology. SARTRE (Similarity Assessment Reasoning for Texas hold'em via Recall of Experience) is the latest result of our ongoing research focused around the investigation into the role of memory in game AI. SARTRE is specifically tailored to play two-player poker, whereas our previous system, CASPER (CASe based Poker playER) was more suited to full-table game play (8 - 10 players) [1, 3]. Two-player poker offers its own unique challenges, where strategies for successful play differ markedly from those employed at a full table [2]. For a description of the rules of Texas Hold'em consult [1, 2].

2 Overview of SARTRE

A human poker player requires information to make their betting decisions. As SARTRE is a computer program, the information required needs to be easily recognised and able to be reasoned about algorithmically. Salient information needs to be identified and used to affect SARTRE'S final decision. While too much information is usually better than not enough, the utilisation of too much information could result in undue complexity which may deteriorate SARTRE'S performance.

The type of information that SARTRE has available at decision time includes items from the following list:

– The betting decisions of each player during the current hand.
– The betting decisions of each player for all hands that occurred previous to the current hand.
– The current stage of the hand.
– The *hole cards* that are only visible by Sartre.
– The *community cards* that are visible by all players.

The authors have hand picked three key factors from the above list to represent *indexed features* that SARTRE uses to determine a solution for a particular case:

1. The previous betting for the current hand.
2. The current strength of SARTRE'S hand given by combining personal *hole cards* with the publicly available board cards.
3. Information about the state of the current community cards, called the *texture of the board*.

Qualitative feature descriptions have been favoured over quantitative descriptions as they are more likely to be used by an expert, human player. Each case feature is described in more detail below, including the representation we have chosen to implement for the SARTRE system.

2.1 The previous betting for the current hand

The type of betting that can occur at each decision point in a hand consists of a fold *(f)*, check/call *(c)*, or bet/raise *(r)*. A combination of these symbols corresponds to all the decisions made during a particular hand.

As SARTRE is specifically designed to play only `heads-up` poker the number of betting patterns that can occur is drastically reduced compared to the combination of betting patterns that can occur at a table with ten players. An example betting pattern is presented and analysed below. The total bets allowed to be contributed by each player during each round is capped at four:

`rrc-r`

This particular example represents a feature that is used by SARTRE during the flop. The betting that occurred during the *pre-flop* is separated from the betting on the *flop* by a hyphen. This betting string can be described by the following situation:

– SARTRE is in the *small blind* (dealer) and makes a forced bet of 0.5. The opponent is in the *big blind* and makes a forced bet of 1.
– SARTRE is the first to act *pre-flop* and decides to *raise*. SARTRE has now committed a total of 2 bets.
– SARTRE'S opponent *re-raises* by committing another 2 bets to the pot.
– SARTRE calls 1 more bet completing the *pre-flop* betting and leaving 6 total bets in the pot.

– As SARTRE is the dealer he acts last on all *post-flop* betting rounds. SARTRE'S opponent has made 1 bet on the *flop* and it is now SARTRE'S turn to make a decision.

The above example indicates a lot of information is contained within the betting pattern. We have chosen to represent each betting pattern as a path within a betting tree. A betting tree succinctly enumerates all betting combinations up until a certain point in the hand. A path within this tree represents the actual decisions that were made by each player during this hand. This is represented graphically in Fig. 1.

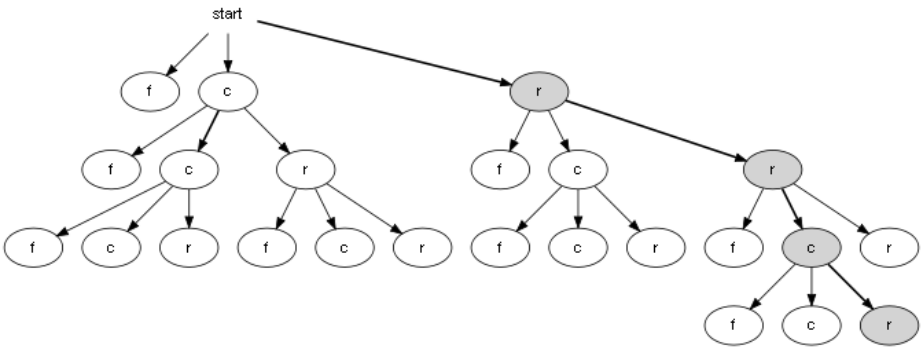

Fig. 1. A tree that describes betting decisions for two players during a hand of Texas Hold'em Poker. The highlighted nodes are the actual decisions that were made by each player.

Given this representation we can calculate the similarity between two separate trees (a target tree and a source tree) by comparing the betting path within each tree. If the betting path in the target tree is exactly the same as the betting path within the source tree a similarity value of 1.0 is assigned. Currently, Sartre will simply assign a value of 0.0 to any betting paths that are not exactly similar, however, we plan to investigate less stringent approaches for future implementations. For example, if one betting path mostly resembles that of another, with a small number of variations, a similarity value close to (but less than) 1.0 could be assigned.

2.2 The current hand category

The second case feature used to determine a betting action is a qualitative category describing SARTRE'S personal hand. During the *pre-flop* SARTRE'S hand simply consists of his personal *hole cards*, whereas for the *post-flop* stages of play SARTRE'S hand is constructed by combining his *hole cards* with the publicly available *community cards*, the best 5 card combination is used.

SARTRE'S best 5 cards are mapped to a category that describes the hand. The classic hand categories in poker include *no-pair, one-pair, two-pair, three-of-a-kind, straight, flush, full-house, four-of-a-kind* and finally a *straight-flush*. Each category has a greater strength than the previous one, where a *straight-flush*, consisting of the cards `Ten, Jack, Queen, King, Ace`, represents the highest rank possible (i.e. a *Royal Flush*).

During the *flop* and the *turn* all the community cards have yet to be dealt and therefore a player's hand has the ability to improve from one category to another, depending on which card is drawn next. It is therefore too simplistic to only consider the current hand category, so further classification is required for hands with the potential to improve. These types of hands are called *drawing hands* (in poker terminology). SARTRE considers two types of drawing hands: *flush draws* & *straight draws*. An example mapping is illustrated in Fig. 2.

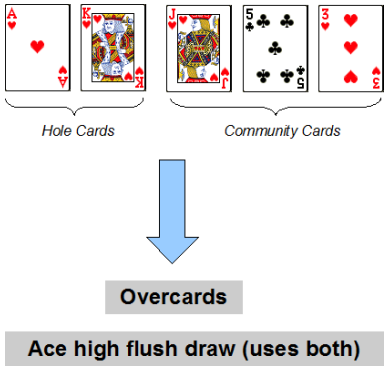

Fig. 2. Mapping a combination of five cards to a category that represents the current hand rank and the drawing strength of this hand.

The hand categories SARTRE uses to classify cards were decided upon by the authors. Fig. 2. shows a combination of two categories: *overcards* + *ace-high-flush-draw-uses-both* i.e. no pair has been made, but both *hole cards* have a higher rank than the community cards and this hand has the potential to become a *flush*.

Currently a simple rule-based system is used to decide which category a combination of cards belongs to. Similarity for this feature is currently either 1.0 when the category of the target case is exactly that of the source case, otherwise it is 0.0 when the categories are distinct.

2.3 The texture of the board

The final indexed feature attempts to summarise the state of the community cards without considering the *hole cards* of a player. The *texture of the board*

refers to salient information a human poker player would usually notice about the public cards, such as whether a *flush* is possible. Once again a set of qualitative categories were hand-picked by the authors to map various boards into. Some categories used by SARTRE'S current implementation that refer to flush and straight possibilities are *Is-Flush-Possible* (where three cards of the same suit are showing), *Is-Flush-Highly-Possible* (where four cards of the same suit are showing) & *Is-Straight-Possible* (where three consecutive card values are showing), *Is-Straight-Highly-Possible* (where four consecutive card values are showing).

If two boards are mapped into the same category they are given a similarity value of 1.0, whereas boards that map to separate categories have a similarity of 0.0.

2.4 SARTRE'S Case-Base

SARTRE'S case-base is generated by analysing the game logs of previous AAAI Computer Poker Competitions[1]. The current version of SARTRE uses approximately 250,000 cases for each stage of the game (*pre-flop, flop, turn, river*). Results against other computerised opponents will soon be available as we plan for SARTRE to compete in the upcoming Computer Poker Competition to be held at IJCAI '09.

3 Future Work

Work on the SARTRE system is still in an early phase and there is much room for investigation and improvement:

1. Experimental results are required. At present we have not had time to extensively measure SARTRE'S performance
2. Currently SARTRE'S similarity metrics are too stringent. Further work is required to improve the calculation of similarity values.
3. SARTRE uses no opponent modelling capabilities at present. We plan to augment SARTRE with a CBR opponent modelling system and assess the impact on performance.

References

1. Rubin, J., Watson, I.: Investigating the Effectiveness of Applying Case-Based Reasoning to the Game of Texas Hold'em. Proceedings of the Twentieth International Florida Artificial Intelligence Research Society Conference. AAAI Press. 417–422 (2007)
2. Sklansky, D.: The Theory of Poker. Two Plus Two Publishing, Las Vegas, Nevada (2005)
3. Watson, I., Rubin, J.: CASPER: A Case-Based Poker-Bot. AI 2008: Advances in Artificial Intelligence, 21st Australasian Joint Conference on Artificial Intelligence. Springer (5360). Lecture Notes in Computer Science. 594–600 (2008)

[1] http://www.cs.ualberta.ca/ pokert/