Self Fuzzy Learning

Tristan Cazenave

LAFORIA-IBP
case 169
Université Pierre et Marie Curie
4, place Jussieu
75252 PARIS CEDEX 05
Tel: 44 27 70 10
Fax: 44 27 70 00
cazenave@laforia.ibp.fr

Abstract

This paper explains a method to learn from fuzzy explanations. This method has been applied to learn strategic rules in the Game of Go. Explanation Based Learning uses a representation of knowledge mainly based on the predicate logic. My goal is to extend this method of learning to systems using fuzzy logic. It is not useful to have gradual knowledge in order to learn tactical knowledge, but it becomes necessary when learning strategic knowledge. Strategic knowledge is fuzzy by nature. I give a method to learn using fuzzy explanations. This method is supported by an example of the learning of a rule in the game of Go. It shows how this method can be applied in complex domains.

Key words : Explanation Based Learning, Fuzzy logic, Strategy, Game of Go.

1 Introduction

When a domain theory exists, a method to deductively learn rules has been developed : Explanation Based Learning (EBL) [Mitchell 1986] [Dejong 1986]. This learning method is particularly useful in the domain of games. Games have a strong domain theory. Many projects using EBL to learn tactical plans have been developed [Minton 1984] [Puget 1987] [Tadepalli 1989] after the initial work of Jacques Pitrat on Chess [Pitrat 1976].

This article explains a method to learn using fuzzy explanations. I have developed a systems which learns tactical plans in the game of Go. It learns using explanations on the problems is has solved [Cazenave 1996b]. My system also gives strategic explanations on its moves [Cazenave 1996a]. Explanation Based Learning uses predicate logic to represent knowledge. The goal of this paper is to extend it to knowledge representation with fuzzy logic. Representing gradual knowledge is not necessary from a tactical point of view, but it becomes necessary on a strategic point of view. Fuzzy logic has already been applied to search, and especially to Chess [Junghanns 1995], but it has been used to control search. My purpose is not to control search but to create automatically large fuzzy knowledge bases of rules.

In a first part, I show why a fuzzy knowledge representation is adapted to the strategic knowledge of Go players. In a second part, I explain how this fuzzy knowledge can be used by a self fuzzy learning system to develop itself from a small set of initial rules. I finish with the possible extensions of my system.

2 A Fuzzy Strategy

Strategic knowledge in games are about long term goals. In games such as Chess and Go, the high number of possible moves makes it impossible to forecast in a long term the consequences of the moves played. A solution to this problem is to have a gradual achievement of long term goals. It enables to know if a move makes the goal easier or harder to achieve.

This is particularly true for strategy in the game of Go. The ultimate goal of a player is to make live the more stone on the board. However, in the middle game, most of the groups of stones are in an uncertain state, and the evolution of this state cannot be precisely foreseen. It is very useful in such a case to have a fuzzy evaluation of their states and of the evolution of this state when playing different moves.

Definition : A group of stones is a set of stones of the same color which cannot be disconnected.

Example :

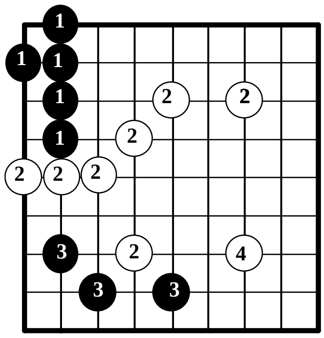


Figure 1

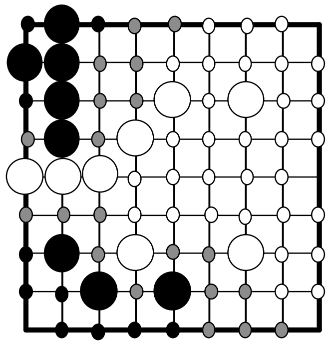


Figure 2

Stones of the same group have the same number in Figure 1.

There are many attributes for a group. Table 1 gives a list of the attributes related to a group used in my system.

Number of won life bases
Number of unsettled life bases
Number of won eyes
Number of unsettled eyes
Number of connectable friend intersections
Number of connectable intersections
Number of stones
Number of connections to living friends

Table 1

Each of these attributes contributes to the final goal of the game which is to make the group live. These contributions are less or more graduals. They are represented in Figure 1 to 8. The vertical axis always represents the degree of life of the group, between 0 and 1.

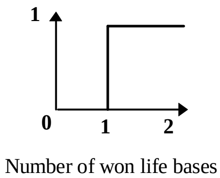


Figure 3

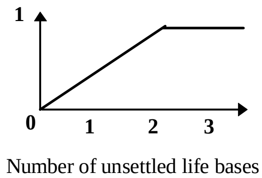


Figure 4

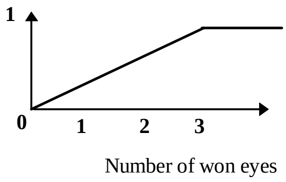
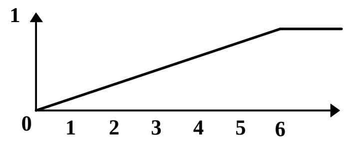
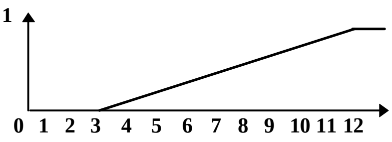


Figure 5



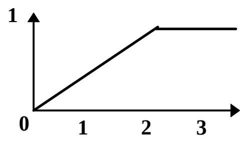
Number of unsettled eyes

Figure 6



Number of connectable friend intersections

Figure 7



Number of connections to living friends

Figure 8

In Figure 2, the intersections connectable to a white group are filled with a white point. The intersections connectable to a black group are filled with a black point. The intersections connectable both to a white and a black group are filled with a gray point. Table 2 gives an evaluation of the attributes for the four groups of Figure 1.

Attributes\Groups	1	2	3	4
Number of won life bases	0	0	0	0
Number of unsettled life bases	1	0	0	0
Number of won eyes	1	0	0	0
Number of unsettled eyes	1	0	0	0
Number of connectable friend intersections	3	26	7	11
Number of connectable intersections	11	40	19	18
Number of stones	5	7	3	1
Number of connections to living friends	0	0	0	2

Table 2

I have chosen to never overestimate the degree of life of a group. So I use the attributes as if they had no influence on each other. Thus the degree of life of a group is the maximum of all the degree of life corresponding to each attribute. Table 3 gives the degrees of life corresponding to each attribute for each group and also gives the final degree of life for the groups.

Attributes\Groups	1	2	3	4
Number of won life bases	0	0	0	0
Number of unsettled life bases	0.5	0	0	0
Number of won eyes	0.33	0	0	0
Number of unsettled eyes	0.16	0	0	0
Number of connectable friend intersections	0	1	0.44	0.89
Number of connections to living friends	0	0	0	1
Degree of live of the group	0.5	1	0.44	1
Importance of the group	24	80	32	31

Table 3

The importance of a group is evaluated by the following formula:

$$\text{Importance}_i = 2^* \text{ Number of stones}_i + \text{Number of connectable intersections}_i + \text{Number of connectable friend intersections}_i$$

The importances of the example groups are given in Table 3. The importance of a group is the difference of points at the end of the game between the live of the group and its death.

When the importances and the degrees of life of the groups have been computed, we can evaluate a Go board:

$$\text{Evaluation} = \sum_i (\text{Degree}_i * \text{Importance}_i) - \sum_j (\text{Degree}_j * \text{Importance}_j)$$

with $i \in$ Friends Groups and $j \in$ Enemies Groups.

In the example of Figure 1, if black is the friend color, the evaluation of the position gives:

$$\text{Evaluation} = 0.5*23 + 0.44*32 - 1.0*80 - 1.0*31 = 11.5 + 14.1 - 80 - 31 = -85.4$$

This evaluation means that black is probably going to lose the game by 43 points. This analysis is compatible with the analysis of Go expert players. This evaluation function has been tested on numerous Go boards and it gives a good approximation of the evaluation of a position.

3 Self Fuzzy Learning

This section shows how fuzzy rules can be used by a system to improve itself automatically by forecasting better and better the consequences of its moves.

3.1 Deduce the consequences of a move

For each move, the system has an associated set of goals the move achieves.

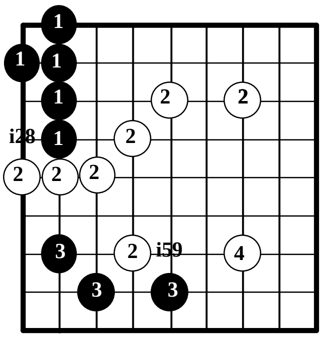


Figure 9

The two moves we are examining in the board of Figure 9 are the black moves in i28 and i59. Table 4 gives the outcomes of the black move in i28 and Table 5 gives the outcomes of the black move in i59.

Attributes\Groups	1	2	3	4
Number of won life bases	+1	0	0	0
Number of unsettled life bases	-1	0	0	0
Number of won eyes	+1	0	0	0
Number of unsettled eyes	-1	0	0	0
Number of connectable friend intersections	0	0	0	0
Number of connections to living friends	0	0	0	0

Table 4

Attributes\Groups	1	2	3	4
Number of won life bases	0	0	0	0
Number of unsettled life bases	0	0	0	0
Number of won eyes	0	0	0	0
Number of unsettled eyes	0	0	+1	0
Number of connectable friend intersections	0	-4	0	-1
Number of connections to living friends	0	0	0	-1

Table 5

If the board is evaluated after the two black moves, there is a variation of +12 points for the black move in i28 and a variation of +11 points for the black move in i59. The system will choose the black move in i28.

3.2 Explain fuzzy deductions

The explanation consists in giving the reasons why a move was chosen. In the case of the black move in i28, the reason of selecting the move was that it allows to transform an unsettled life base in a won life base. But also that all the other parameters influencing the live of the group have a fuzzy contribution lesser than 0.5 (Cf. Table 3). The black move in i28 has therefore a value of 0.5*24=12 points. The explanations of this deduction are given in Table 6.

Color of group 1 = Black Number of won life bases of group 1 = 0 Number of unsettled life bases of group 1 = 1 Number of won eyes of group 1 < 2 Number of unsettled eyes of group 1 < 2 Number of connectable friend intersections of group 1 < 4 Number of connections to living friends of group 1 = 0 Importance of group 1 = 24 The black move in i28 appends a life base to group 1 12 = 24 * 0.5
The Black move in i28 has a value of 12 points

Table 6

3.3 Generalize the explanation

When the explanation is done, we can generalize the explanation to allow it to apply in many more case. The main mechanism of generalization is the replacement of instanciated variables by constants. The variables are always beginning with a '?' in my system. The generalized explanation of the black move in i28 is given in Table 7.

Color of group ?n = ?color Number of won life bases of group ?n = 0 Number of unsettled life bases of group ?n = 1 Number of won eyes of group ?n < 2 Number of unsettled eyes of group ?n < 2 Number of connectable friend intersections of group ?n < 4 Number of connections to living friends of group ?n = 0 Importance of group ?n = ?n1 The black move in ?i appends a life base to group ?n ?n2 = ?n1 * 0.5
The ?color move in ?i has a value of ?n2 points

Table 7

This generalized explanation is transformed in a strategic rule. this strategic rule is very general and can be applied in many more boards than the example board on which it was learned. The rules created by the system are used to learn other rules. The systems bootstraps itself by creating more and more rules until no more interesting rule can be created.

4 Conclusion

I have described a method to automatically create strategic fuzzy rules in the game of Go. This method can be used to bootstrap a large base of fuzzy rules beginning with a small set of rules. It create a large set of valid, useful and general rules using only the simple definition of the strategic goals of the system. My system uses strategic fuzzy rules and plays to an international level [Pettersen 1994]. This learning algorithm can be applied to other domains than the game of Go. It is adapted to very complex domains where the important goals are better represented using gradual knowledge. In domains where it is impossible to compute directly if a goal is achievable because of the combinatorial explosion of the search.

References

[Cazenave 1996a] - T. Cazenave. *Constraint-Based Explanations in Games.* IPMU96, Grenade, 1996.

[Cazenave 1996b] - T. Cazenave. *Learning to Forecast by Explaining the Consequences of Actions.* Machine Learning, Forecasting and Optimization 96, Madrid, 1996.

[Dejong 1986] - G. Dejong, R. Mooney. *Explanation Based Learning : an alternative view.* Machine Learning 2, 1986.

[Junghanns 1995] - A. Junghanns. *Search with Fuzzy Numbers.* 4th IEEE International Conference on Fuzzy Systems, Yokohama, Japan, 1995.

[Minton 1984] - S. Minton. *Constraint-Based Generalization - Learning Game-Playing Plans from Single Examples*. Proceedings of the Fourth National Conference on Artificial Intelligence, 251-254. Los Altos, William Kaufmann, 1984.

[Mitchell 1986] - T. M. Mitchell, R. M. Keller, S. T. Kedar-Kabelli. *Explanation-based Generalization : A unifying view*. Machine Learning 1 (1), 1986.

[Pettersen 1994] - E. Pettersen. *The Computer Go Ladder*. World Wide Web page: http://cgl.ucsf.edu/go/ladder.html, 1994.

[Pitrat 1976] - J. Pitrat. *Realization of a Program Learning to Find Combinations at Chess*. Computer Oriented Learning Processes, Simon J. Ed., Noordhoff, 1976.

[Puget 1987] - J. F. Puget. *Goal Regression with Opponent*. Progress in Machine Learning, Sigma Press, Wilmslow, 1987.

[Tadepalli 1989] - P. Tadepalli. *Lazy Explanation-Based Learning: A Solution to the Intractable Theory Problem*. IJCAI 1989, pp. 694-700, 1989.