Concept Paper - Applications of deductive machine learning to Game playing strategy

OWOMUGISHA ISAAC	15/U/12351/PS
MUGUMYA LEONALDS	14/U/9560/PS
MUWONGE JULIUS	15/U/8595/PS
OCHAKA SAMUEL	14/U/25135

1 Introduction

The goal of deductive machine learning is to provide computers with the ability to automatically learn a behavior that provably satisfies a given high-level specification. As opposed to techniques that generalize from incomplete specifications, deductive machine learning starts with a complete problem description and develops a behavior as a particular solution. The aim is to use techniques from deductive machine learning to design an algorithm or agent that can learn a winning strategy for this type of game.

Machine learning is an important branch of artificial intelligence that has in recent years gained a lot of attention due to its applications in big data analysis and game playing as evidenced by AlphaGo, the first AI to defeat a professional in the ancient game of Go. According to [1], the various types of machine learning can be classified as either deductive or inductive. Inductive learning involes learning a general function or rule from specific input-output pairs. Deductive machine learning (sometimes called analytical machine learning) involves going from a known general rule to a new rule that is logically entailed, but is more efficient because it allows more efficient processing. In this research, we will focus on deductive machine learning.

2 Problem statement

The problem we aim to solve in this research project can be thus be summarized as follows: Given the rules of a game and a specification of a winning criteria, design an agent that can learn a winning strategy for this game using techniques from deductive machine learning.

In each game, there is a goal state which is the winning position (or possibly a draw). The term "strategy" refers to the highest level means that the AI will use to accomplish this overall goal.

3 Background and Scope

The kinds of games we shall focus on for this research project are 2 player games of perfect information such as chess, checkers and Go. A game is said to have perfect information if each player, when making any decision is perfectly informed of all the events that have previously occurred.

Techniques commonly used to solve games include brute-force methods, the minimax algorithm and minimax with alpha-beta pruning. For simple games such as tic-tac toe, these methods can yield an answer easily. However, in more complicated games, the number of possible solutions are too vast for even computers to brute force and sometimes the search trees are so big that even alpha-beta pruning is not effective. For example, in [2], computer scientist Claude Shannon estimated that the number of possible positions in chess exceedded 10^{120}, far greater than the number of atoms in the known universe. In this research project, we hope to explorer more efficient methods.

Game playing is an important area of research in artificial intelligence because games are the perfect platform for developing and testing AI algorithms quickly. In addition, the techniques used in solving games and learning winning strategies can be applied to more general settings that can be used to address some of society's toughest and most pressing problems such as climate modelling and complex disease analysis.

4   Objectives

- To explore different methods used in deductive machine learning and how they can be used to learn a winning strategy for a game.

- To design deductive machine learning algorithms (or use existing ones) which learn a winning strategy for a game.

5   Methodology

We shall use general first-order logic theories to represent the rules and winning criteria of the games we shall consider here (this is known as the knowledge representation). From there we shall attempt to apply knowledge-based techniques such as explanation-based learning and relevance based learning; we shall also attempt to show how these deductive methods can be combined with inductive methods such as neural networks to achieve better results.

Below is an example of how the winning criteria tic-tac toe can be represented using first-order logic.
Consider a board layout numbered such that the first row is "1, 2, 3", second row "4, 5, 6" and third row "7, 8, 9":The predicate ordered_line is used to represent the fact that if a series of x's and o's are put in the positions of its operands, then a winning position has been attained:
$ordered\_line(1,2,3), ordered\_line(4,5,6), ordered\_line(7,8,9), ordered\_line(1,4,7),$
$ordered\_line(2,5,8), ordered\_line(3,6,9), ordered\_line(1,5,9), ordered\_line(3,5,7).$
With the above facts, we can use the following rule to find out if a line is ordered:
$line(A,B,C) \Leftarrow ordered\_line(A,B,C) \vee ordered\_line(A,C,B) \vee ordered\_line(B,A,C)$
$\vee ordered\_line(B,C,A) \vee ordered\_line(C,A,B) \vee ordered\_line(C,B,A).$

Once a game has been represented in such a way, we shall then attempt to apply the techniques highlighted above to find a winning strategy.

References

[1] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall Series in Artificial Intelligence, Prentice Hall, 2010.

[2] C. E. Shannon, "Programming a computer for playing chess," *Philos. Mag. (7)*, vol. 41, pp. 256–275, Mar. 1950.