My First LaTeX Document

Mohammad Ibrahim Khan

December, 2024

Contents

List of Figures

List of Tables

1 Background and Context

1.1 Chess

Chess is an unsolved game. A game dating back to the 7th century [REFERENCE] as a game of strategy and tactics [REFERENCE]. Today it is a game seen as a benchmark for skill and intelligence, played by people in their millions.

1.2 Chess Engines

Since 1997 when IBM's Deep Blue beat Kasparov, a world champion, chess engines have been a popular topic of research. Chess engines analyse millions of positions per second in order to defeat the best human players. However, even with technological advancements since 1997, chess is still unsolved. According to the Shannon number, there are 10^{120} possible positions in chess [REFERENCE], making it unviable to generate all possible positions and evaluate them.

Computer scientists have implemented a variety of ways in order to reduce this search space which will be discussed in subsequent sections.

1.3 Search Algorithms

Chess engines most commonly a minimax search algorithm, where every node is a position in the game and the legal moves create the next layer of nodes. Ideally, the engine would search till the end of the tree, thus always finding the best move. However, due to the very large search space, this is not feasible. Therefore, engines implement a number of different techniques to reduce the search space, including alpha-beta pruning and iterative deepening.

1.4 Machine Learning in Chess

Many researchers have used machine learning techniques to improve the performance of chess engines. These techniques include classification techniques like Neural Networks and Naive Bayes as well as clustering techniques like K-means as well as reinforcement learning techniques like [????Q-learning and Deep Q-learning.]. Some have even used Natural Language Processing techniques.[1] These techniques are usually used in conjuction with traditional search algorithms.

1.5 Motivation

The primary motivation for this research is to explore how Naive Bayes can be implemented in a chess engine. Naive Bayes is a simple classification technique which is computationally lightweight in comparison to other techniques like Neural Networks, which may be ideal for quick predictions in settings with limited computational power, like mobile devices.

2 Literature Review

2.1 Minimax and Alpha-Beta Pruning

The concept of Minimax was first proposed by Shannon in 1950[2]. A zero-sum games is where "the loss of one player is the gain of the other"[3]. Minimax works on the principle of zero-sum games and assumes that the opponent will always make the best move. The algorithm recursively alternates between the maximising player and the minimising player, until a terminal node is reached. The terminal node is then evaluated using a heuristic function. [MAYBE ADD FIGURES for MINIMAX AND PSEUDOCODE]

Alpha-Beta pruning is a an approach used to decrease the number of nodes evaluated by the minimax algorithm. It does this by not evaluating nodes that would not affect the final outcome of the minimax algorithm. It introduces two new values α and β where α represents the maximum value that can be attained and β represents the minimum value that can be attained. If the value of a node

is less than α or greater than β, the tree does not need to be traversed further. [MAYBE ADD PSEUDOCODE]

2.2 Naive Bayes

Naive Bayes, sometimes also known as Idiot Bayes or Simple Bayes, is a simple classification algorithm that utilises Bayes' theorem (Equation 1). It is reffered to as naive as it assumes that each input variable $X_1, X_2, ..., X_n$ is conditionally independant given the class [4]. Despite this assumption not being true in most cases, Naive Bayes has still been shown to work well. This assumpption allows probability distributions to be efficiently represented as the product of the individual probabilities of the input variables (Equation 2) [4].

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{1}$$

$$P(X_1, X_2, ..., X_n, C) = P(C) \cdot \prod_{i=1}^{n} P(X_i|C) \tag{2}$$

References

[1] I. Kamlish, I. B. Chocron, and N. McCarthy, "SentiMATE: Learning to play Chess through Natural Language Processing."

[2] Y. Xie, W. Gao, Z. Dai, and Y. Li, "Research and Improvement of Alpha-Beta Search Algorithm in Gobang,"

[3] A. Plaat, "Research Re: Search & Re-search."

[4] D. Lowd and P. Domingos, "Naive Bayes models for probability estimation,"